

**DISEÑO E IMPLEMENTACIÓN DE SISTEMA DE INTERPRETACIÓN Y
TRADUCCIÓN DE GESTOS ASOCIADOS A PREGUNTAS, NECESIDADES Y
SALUDOS BÁSICOS DEL LENGUAJE DE SEÑAS COLOMBIANO.**



**UNIVERSIDAD DE CORDOBA
FACULTAD DE INGENIERÍAS
PROGRAMA DE INGENIERÍA DE SISTEMAS Y TELECOMUNICACIONES
Montería, Colombia
2015**

**DISEÑO E IMPLEMENTACIÓN DE SISTEMA DE INTERPRETACIÓN Y
TRADUCCIÓN DE GESTOS ASOCIADOS A PREGUNTAS, NECESIDADES Y
SALUDOS BÁSICOS DEL LENGUAJE DE SEÑAS COLOMBIANO.**

Integrante:

Jaime Luis Ballesta Pérez

Tesis para optar el título de Ingeniero de Sistemas

Asesor: Msc Samir Castaño



**UNIVERSIDAD DE CÓRDOBA
FACULTAD DE INGENIERÍAS
PROGRAMA DE INGENIERÍA DE SISTEMAS Y TELECOMUNICACIONES**

Montería, Colombia

2015

Tabla de contenido

1. OBJETIVOS.....	1
1.1. Objetivo general	1
1.2. Objetivos específicos	1
2. INTRODUCCIÓN.....	2
2.1. Ambientación	2
2.2. Problemática.....	4
2.3. Estado del arte.	5
2.3.1. Contexto internacional.....	5
2.3.2. Contexto nacional.....	9
2.4. Justificación.....	12
3. MARCO TEÓRICO Y MARCO CONCEPTUAL	13
3.1. Marco Teórico	13
3.1.2. Visión artificial	14
3.1.3. Algoritmo de reconocimiento Adaboost	14
3.1.4. Interfaz de usuario natural (NUI).....	15
3.1.5. Kinect Studio v2.0	16
3.1.6. Visual Gesture Builder	17
3.1.7. Percepción de profundidad	19
3.1.8. Mapa profundidad Kinect	20
3.1.9. Windows Presentation Foundation (WPF).....	23
3.2. Marco conceptual	24
3.2.1. Discapacidad	24
3.2.2. Sordomudo	24
3.2.3. Visión artificial.....	24
3.2.4. Interface natural de Usuario.....	25
3.2.5. NET.....	25
3.2.6. Lenguaje XAML	25
3.2.7. Programación declarativa	25
3.2.8. Sensor Kinect	26
3.2.9. Visual Studio 2013	29
3.2.10. Lenguaje de programación C#.....	29
4. METODOLOGÍA	30

4.1.	Tipo de investigación	30
4.2.	Línea de investigación	30
4.3.	Población Objetivo	30
4.4.	Fases del proyecto	31
4.4.1.	Fase 1: Análisis.....	32
4.4.2.	Fase 2: Diseño	34
4.4.3.	Fase 3: Desarrollo.....	35
4.4.4.	Fase 4: Pruebas	37
4.5.	Metodología de desarrollo del producto	39
4.5.1.	Análisis.....	40
4.5.2.	Diseño	40
4.5.3.	Desarrollo	41
4.5.4.	Pruebas	41
5.	Desarrollo.....	42
5.1.	Arquitectura del sistema	42
5.2.	Diseño del sistema.....	48
5.2.1.	Especificación de requerimientos	49
5.2.2.	Objetivos del sistema.....	49
5.2.3.	Diagrama de casos de uso	51
5.2.4.	Diagramas de secuencia.....	54
5.2.5.	Diagrama de actividades.....	56
5.2.6.	Requisitos de información	60
5.2.7.	Documentación de los actores	61
5.2.8.	Documentación de los casos de uso	62
5.2.9.	Requisitos no funcionales	72
5.2.10.	Diagrama de componentes.....	73
5.2.11.	Diagrama de clases	74
5.2.12.	Pruebas y análisis	75
6.	Conclusiones	86
7.	Referencias bibliograficas.....	87
	ANEXO.....	89

Tabla de figuras

Figura 1. Grabando datos a través del Kinect Studio.....	17
Figura 2. Alimentando la base de datos de gesto en el VGB.	18
Figura 3. Mapa de profundidad.	20
Figura 4. Transformación a escala de grises.....	21
Figura 5. Skeletal Tracking	22
Figura 6. Sensor Kinect v2.0.....	27
Figura 7. Componentes del Kinect v2.0.	28
Figura 8. Población Objetivo.....	31
Figura 9. Fases del proyecto.....	32
Figura 10. Modelo de desarrollo.	39
Figura 11. Arquitectura del sistema.....	43
Figura 12. Grabando clips de video.....	45
Figura 13. Conversión de clips de video.	46
Figura 14. Enseñándole al sistema que gestos serán válidos	47
Figura 15. Diagrama de la estructura del desarrollo.	48
Figura 16. Diagrama de Componentes.	73
Figura 17. Diagrama de clases.....	74
Figura 18. Realización del gesto "Buenos días".	76
Figura 19. Realización del gesto "Buenos días", mal interpretado.	76
Figura 20. Realización del gesto "Hola" por otro usuario.....	78
Figura 21. Realizando gestos que no existe en las bases de datos.	79
Figura 22. Vista preliminar de la nueva interfaz del sistema.	80
Figura 23. Realización del gesto para pasar de una sección a otra.....	81
Figura 24. Gesto para navegar entre las secciones hecho por otro usuario.	82
Figura 25. Realización del gesto "Hola" con 2 usuarios frente al sensor.	83
Figura 26. Gesto para navegar entre secciones con dos usuarios frente al sensor.....	84
Figura 27. Manual de usuario, reconocer el cuerpo.	90
Figura 28. Manual de usuario Gesto "Hola".	91
Figura 29. Manual de usuario Gesto "Adiós".	92
Figura 30. Manual de usuario Gesto "Buenos días".	93
Figura 31. Manual de usuario Gesto "Buenas tardes".	94
Figura 32. Manual de usuario Gesto "Buenas noches".	95
Figura 33. Manual de usuario Gesto "Siguiendo parte 1".	96
Figura 34. Manual de usuario Gesto "Siguiendo parte 2"	97
Figura 35. Manual de usuario Gesto "Anterior parte 1".	98
Figura 36. Manual de usuario Gesto "Anterior parte 2".	99
Figura 37. Manual de usuario Gesto "como estas".	100
Figura 38. Manual de usuario Gesto "donde esta".	101
Figura 39. Manual de usuario Gesto "cuál es tu nombre".	102
Figura 40. Manual de usuario Gesto "Por qué".	103
Figura 41. Manual de usuario Gesto "Que estás haciendo".	104
Figura 42. Manual de usuario Gesto "Por favor".	105
Figura 43. Manual de usuario Gesto "Tengo hambre".	106

Figura 44. Manual de usuario Gesto "Tengo sed".	107
Figura 45. Manual de usuario Gesto "Tengo sueño".	108
Figura 46. Manual de usuario Gesto "Gracias".	109

Tabla de tablas

Tabla 1. Pasos del algoritmo Adaboost.	15
Tabla 2. Realización del gesto.....	50
Tabla 3. Traducción de los Gestos.....	51
Tabla 4. Requisitos de información sobre realización de gesto.	60
Tabla 5. Requisitos de información sobre traducción de gestos.	61
Tabla 6. Información sobre el actor: Usuario.	61
Tabla 7. Información sobre el actor: Sistema.	62
Tabla 8. Documentación del caso de uso ubicarse frente al sensor.....	63
Tabla 9. Documentación del caso de uso reconocer cuerpo	65
Tabla 10. Documentación del caso de uso reconocer el cuerpo.....	66
Tabla 11. Documentación del caso de uso validar gesto.	68
Tabla 12. Documentación del caso de uso identificar gesto.	69
Tabla 13. Documentación del caso de uso traducir gesto.	70
Tabla 14. Documentación del caso de uso mostrar traducción.	71

1. OBJETIVOS

1.1. Objetivo general

- ❖ Desarrollar un sistema de interpretación y traducción de gestos asociados a preguntas, necesidades y saludos básicos del lenguaje de señas colombiano, utilizando interfaces naturales de usuario.

1.2. Objetivos específicos

- ❖ Estudiar el lenguaje de señas o lenguaje de signos colombiano para identificar sus características y aspectos relevantes.
- ❖ Realizar la identificación de gestos asociadas a preguntas, necesidades y saludos básicas del lenguaje de señas Colombiano
- ❖ Estudiar el desarrollo de aplicaciones basadas en el dispositivo de interface natural de usuario KINECT versión 2.
- ❖ Utilizar el Kinect Studio y Visual Gesture Builder, como herramientas para la construcción de un repositorio de gestos basados en movimientos discretos continuos.
- ❖ Desarrollar una aplicación para la captura e identificación de gestos utilizando el dispositivo Kinect versión 2.

2. INTRODUCCIÓN

2.1. Ambientación

La discapacidad es una realidad humana percibida de manera diferente en diversos períodos históricos y civilizaciones. En términos generales se podría decir que la discapacidad Es cualquier restricción o impedimento de la capacidad de realizar una actividad en la forma o dentro del margen que se considera normal para el ser humano.

La discapacidad se caracteriza por excesos o insuficiencias en el desempeño de una actividad rutinaria normal, los cuales pueden ser temporales o permanentes, existen varios tipos de discapacidad entre los cuales encontramos la Discapacidad Física, esta se refiere a la destreza motora o capacidad de física, comprometiendo así la movilidad, la Discapacidad Sensorial, está comprendida por deficiencias visuales, auditivas o de la comunicación, Discapacidad Psíquica, se caracteriza por alteraciones neurológicas y trastornos cerebrales, por último la Discapacidad Intelectual, se evidencia por una disminución de las funciones mentales superiores (inteligencia, lenguaje, aprendizaje, entre otros), así como de las funciones motoras. Esta discapacidad abarca toda una serie de enfermedades y trastornos, dentro de los cuales se encuentra el retraso mental, el síndrome Down y la parálisis cerebral.

Dentro de las discapacidades sensoriales encontramos la discapacidad auditiva, la cual se refiere a la falta o disminución para oír, esto es debido a la pérdida en algún lugar del aparato auditivo ya sea desde el nacimiento o provocado por cualquier circunstancia o debido a alguna enfermedad o, accidente en el transcurso de la vida.

La sordera afecta la comunicación de las personas, a unas más que a otras dependiendo de si nació sordo o perdió la audición después de haber aprendido a hablar y de si tiene algún resto auditivo. Cuando una persona nace sorda o pierde la audición a una edad muy temprana, es difícil que aprenda a hablar pues no puede escuchar los sonidos del idioma oral; sin embargo, si desea intentarlo, se le debe enseñar con un proceso largo y costoso.

Las personas sordas no pueden escuchar, pero pueden ver, por esto, el idioma que usan para comunicarse es la LENGUA DE SEÑAS, una combinación de movimientos manuales, gestos, expresiones faciales, corporales y espaciales que integran su lengua (FENASCOL, 2014)¹. El lenguaje de señas, es una lengua natural de expresión y configuración gesto-espacial y percepción visual (o incluso táctil por ciertas personas con sordo ceguera), gracias a la cual los sordos pueden establecer un canal de comunicación con su entorno social, ya sea conformado por otros sordos o por cualquier persona que conozca la lengua de señas empleada. Mientras que con el lenguaje oral la comunicación se establece en un canal vocal-auditivo, el lenguaje de señas lo hace por un canal gesto-viso-espacial.

Aunque existan estas alternativas actualmente, las personas con esta discapacidad aun dependen de que las demás personas manejen el lenguaje de señas para poder interactuar con ellas, es decir, si las demás personas no manejan este lenguaje de igual forma no podrán interactuar ni establecer comunicación de manera independiente, por esta razón sería conveniente sistematizar este proceso con el fin de permitir la comunicación entre las personas con discapacidad y las personas sin discapacidad independientemente de que conozca o no el lenguaje de señas.

¹ FENASCOL. Las personas sordas. tomado de: http://www.fenascol.org.co/index.php?option=com_content&view=article&id=13&showall=1. Fecha de consulta 2014

2.2. Problemática.

Actualmente, más del 5% de la población mundial, es decir, más de (360 millones de personas) padece pérdida de audición discapacitante (328 millones de adultos y 32 millones de niños), de las cuales un 87 % no aprende a leer ni escribir (Organización Mundial de la Salud, 2015).²

Según el (DANE, 2005), la población general en Colombia ascendía a 41.468.384 personas, de las cuales 2.624.898 presentaban alguna discapacidad permanente, cifra que representaba 6,3% de la población total; es decir, que la tasa de incidencia de la discapacidad era del 6,33%, lo que significa que por cada 100 habitantes 6 tenían algún tipo de discapacidad. Por su parte, el censo arrojó que del total de la población 439.412 eran personas que presentaban algún déficit de audición, lo cual significa que la tasa de prevalencia de la sordera era de 1,1%, es decir que en el país 1 de cada 100 habitantes presentaban dificultades para oír. Así, la sordera se constituye en el tercer factor de limitación para las personas con discapacidad, la cuales representan el 17,4% después de las limitaciones para ver con el 43,2% y caminar o moverse con el 29,5%.³

La mayor parte de la población con discapacidad en el mundo se encuentra excluida del ámbito laboral, así como de otros aspectos de la sociedad, debido a los distintos obstáculos que enfrentan en su vida diaria. El Informe Mundial sobre la Discapacidad de 2011, de la Organización Mundial de la Salud (OMS), y un reciente estudio del 2013 de la Broadband Commission for Digital Development (BCFDD) muestran que los obstáculos se derivan de una política poco inclusiva, con deficiencias en los sistemas de salud y educación y el bajo acceso a los avances de la ciencia y la tecnología como mecanismos para mejorar el acceso a la empleabilidad.

² Organización mundial de la Salud (OMS). Tomado de: <http://www.who.int/mediacentre/factsheets/fs300/es/>. Fecha de consulta 2015

³ Instituto nacional para sordos (INSOR). Tomado de: DANE – Censo 2005.

El informe de la (Organización Mundial de la Salud, 2011) plantea como obstáculos para acceder al campo laboral y otros ámbitos de la vida humana, las políticas y normas insuficientes (su poca o nula aplicación), las actitudes negativas de la sociedad, la prestación insuficiente de servicios (en salud y educación), la financiación insuficiente de los programas de inclusión y la falta de accesibilidad a las tecnologías de la información y las comunicaciones.

2.3. Estado del arte.

En el siguiente estudio se muestran las diferentes herramientas, sistemas y tecnologías realizadas, tanto a nivel nacional como internacional para tratar de ayudar a las personas con discapacidad (Sordomudos) a comunicarse con el resto de la sociedad.

El estudio está enfocado en torno a las propuestas o herramientas tecnológicas que se han desarrollado con el fin de brindarles a los sordomudos una forma de interactuar con las personas del común.

2.3.1. Contexto internacional.

Actualmente en el mundo hay muchas personas con discapacidad auditiva y del habla y ciertamente a futuro existirán más personas con esta limitante, ya que son muchas las causas por las que un ser humano puede nacer con esta discapacidad. Aparte de factores hereditarios, durante el embarazo algunas complicaciones o comportamientos maternos, pueden dar lugar a un bebé con problemas de audición. Si la futura madre llega a ingerir ciertos medicamentos o se enferma de rubeola o gripe entre otras, él bebé puede llegar a sufrir esta discapacidad. En el parto, el hecho de ser prematuro o que el parto se complicado o prolongado pueden predisponer al bebé a la sordera. Además de esto, después del nacimiento patologías como otitis, paperas, sarampión o meningitis pueden llegar a dañar el oído. Entre muchas otras causas que pueden producir esta discapacidad

auditiva y del habla. En nuestra sociedad es mínima la población que está en la capacidad de comunicarse con una persona que presente esta discapacidad (sordomudo), ya que para ello es necesario manejar y entender alguna de los métodos que estos utilizan para comunicarse, entre los cuales el que más se destaca es el lenguaje de señas, y este lenguaje como cualquier otro posee reglas, fonemas, modismos, etc.

Por tal motivo se ha tratado de dar solución o por lo menos de minimizar al máximo esta problemática por ello (Universidad Católica “Nuestra Señora de la Asunción”) lanzó un intérprete de la lengua de señas virtuales.⁴ El software fue desarrollado por la Universidad de East Anglia y la productora británica de software de animación para televisión Televirtual, y consistió en un personaje de animación, que traduce a la lengua de signos británica. El programa fue realizado inicialmente en periodo de pruebas a través de la página web de Deaf Connections, aunque la organización recibió opiniones positivas sobre el mismo. Sin embargo, aún quedan por resolver algunos problemas de compatibilidad del software con ciertos ordenadores. (Sue Moore), responsable de Servicios de Deaf Connections, afirmó que los programadores trabajaron con personas sordas con el fin de incorporar al software los signos específicos del dialecto que se habla en su región.

No obstante es solo una parte de la investigación, ya que en España en el año 2006 se implementó una interface para el reconocimiento de señas gestuales para personas hipoacusicas (Incertis, 2006)⁵ el cual consistió en la identificación de la mano cubierta con un guante de color azul. Los contornos de la mano fueron extraídos de la imagen a través de un segmento mediante el modelo de color HSV. Los resultados eran comparados con un diccionario de signos dependiendo de la distancia y así realizaban el reconocimiento de las letras del alfabeto.

Por otro lado (Serradilla, Rodríguez, & Francisco, 2003) presentaron una propuesta para el reconocimiento automático del lenguaje de signos a partir de datos tridimensionales

⁴ TECNOLOGIA PARA DISCAPACITADOS, Teoría y Aplicación a la Informática(Universidad Católica “Nuestra Señora de la Asunción”)

⁵ Incertis I. (2006). Hand gesture recognition for deaf people interfacing in the 18th international conference on pattern recognition, (ICPR’06) IEEE.

correspondientes a posición y orientación de la mano capturados mediante el guante de datos P5. Dada la naturaleza continua y la posibilidad de descomponer los signos en sus constituyentes primarios, se plantea el uso de una arquitectura con diversos módulos encargados de resolver el problema de segmentación, pre clasificar cada uno de los constituyentes primarios mediante redes neuronales y/o técnicas basadas en similitud estadística y finalmente, recomponer el signo e identificarlo.

Independientemente a estos avances en la materia, (L., 2009) desarrolló un sistema para el reconocimiento del alfabeto dactilológico⁶, en este se realizó un umbralado por el método de Otsu para segmentar la mano, se extrajo el contorno, Y el código de contorno resultante se ponderó, Se clasificó el símbolo y se comparó mediante un algoritmo de votación Para obtener el significado del signo en caso de ser reconocido.

Estos intentos de resolver esta problemática no se han estancado en el tiempo, ya que después de estos adelantos se empezaron a introducir nuevas herramientas tecnológicas para mejorar el reconocimiento de los gestos a raíz de esto, (Li, 2012), se propuso el reconocimiento de gestos manuales utilizando KINECT⁷. En esta propuesta se realizó primero de la extracción de las manos utilizando el dispositivo KINECT, después se realizaron el cálculo de la envolvente convexa usando el algoritmo de Graham. A partir de ahí clasificaron el reconocimiento de cada seña por medio del conteo de dedos, por el dedo específico detectado y por una comparación de lectores década dedo.

Con la inclusión del dispositivo de reconocimiento de gestos Kinect en el campo de la problemática, surgieron más propuestas, entre las cuales también encontramos (Du & To, 2012) diseñaron un sistema para el Reconocimiento de gestos manuales utilizando KINECT⁸. En esta propuesta se obtuvo una imagen del mapa de profundidad del dispositivo KINECT para segmentado del área de interés y a partir de ella se continuo el

⁶ Razo L. (2009). sistema para el reconocimiento del alfabeto de dactilológico (tesis de maestría), centros de investigación en computación del instituto politécnico nacional.

⁷ Li Y. (2012). hand gesture recognition using Kinect en 3rd international conference on software engineering and service science IEEE.

⁸ Du H. & To T. (2012). Hand gesture recognition using Kinect, Boston University.

procesamiento usando tonos de gris, a partir de ahí se eliminó por el fondo y se aisló la mano, se eliminó el ruido usando un filtro, después se extrajo el contorno de la mano y se realizó una aproximación poligonal a la línea del contorno con el fin de encontrar los defectos de convexidad. Los defectos se facilitaron para detectar las puntas de los dedos. El reconocimiento se hizo por medio del conteo de dedos y los resultados reportados la detección de cinco señales de la mano.

A partir de punto de investigación se han desarrollado nuevos métodos u opciones para tratar de resolver este problema, entre los cuales encontramos que en el año 2010 en él (Chapala, Instituto Tecnológico Superior (ITS) de, 2010), se desarrolló un traductor de lenguaje de idioma español a lenguaje de señas. El software: “Intérprete de Señas”, traduce el lenguaje español en lenguaje de señas mexicanas para las personas con discapacidad auditiva, es decir, “la persona que quiere comunicarse, escribe las palabras en un teclado y mediante una persona que aparece en la pantalla, las palabras se traducen en señas para facilitar la comunicación con la persona que tiene discapacidad auditiva”⁹.

En contraste a los desarrollos previos, Un grupo de cinco estudiantes de Ingeniería de la Universidad de La Matanza (UNLaM) creó **Onis**, un software para Windows que utiliza la cámara de Kinect y traduce en pantalla el lenguaje de señas argentino (LSA). De esta manera, permite establecer una comunicación entre personas sordas con sus allegados que no sepan el lenguaje; El desarrollo de este sistema tiene como base la interacción de dos funciones, la primera es un traductor, donde el usuario realiza una seña y del otro lado llega texto, mientras que en el segundo modo alguien escribe y representan en el monitor las señas mediante imágenes. Una tercera funcionalidad es el módulo de chat. (Bermúdez, Garrido, Canella, Juárez, & Argentina, 2013).

Por otro lado, en la ciudad de México, Fue desarrollado el proyecto "Intérprete de Lenguaje de Señas", el cual fue realizado en el Centro de Investigación y Tecnología Aplicada (CITA), que forma parte del Parque de Innovación y Transferencia de

⁹ Viviendo mi ciudad. Desarrollan software para traducir el español hablado a señas de sordo mudos. Tomado de: http://www.viviendomiciudad.com/index.php?option=com_content&view=article&catid=29:discapacidad&id=9738:its-de-chapala-obtiene-primer-lugar-en-desarrollo-de-software&Itemid=1005. Fecha de consulta 2014

Tecnología (PIT2); La base principal del proyecto fue el procesamiento de imágenes para el reconocimiento de patrones. Así, cuando una persona se ubicara frente al intérprete de lenguaje de señas y realizara algún movimiento con su cuerpo, el sistema sería capaz de reconocer esos movimientos para realizar la traducción de los signos.

En su elaboración se utilizaron tecnologías de software como Language Integrated Query (LINQ), un proyecto de Microsoft, tecnología que permitió implementar algunos modelos conocidos del reconocimiento de imágenes con mayor sencillez.

También se empleó la tecnología de Microsoft Kinect Research Software Development Kit (SDK) que en la actualidad se encuentra limitada a la industria del entretenimiento, pero en un futuro pudiera ser empleada como un dispositivo de entrada general, tal como el teclado y el ratón. (Microsoft, 2012)¹⁰

2.3.2. Contexto nacional.

En cuanto a los sistemas, y herramientas tecnológicas para facilitar la interacción y comunicación entre las personas sordomudas y las personas del común, es decir hablantes, se ha desarrollado muy poco en Colombia, a pesar de que como en la mayoría de países del mundo, nuestra comunidad con discapacidad auditiva va en ascenso ya sea por aspectos genético al nacer, o por circunstancias fortuitas durante el transcurso de la vida.

Precisamente uno de estos pocos avances se dio en el año 2013 en nuestro país y más explícitamente en Envigado. Los estudiantes, (Daniel Betancur Betancur, Mateo Vélez Gómez & Alejandro Peña Palacio. Escuela de ingenierías de Antioquia, 2013) Desarrollaron de un sistema integrado de hardware y software, para el reconocimiento automático del lenguaje dactilológico de señas utilizado por personas con este tipo de discapacidad. El hardware está compuesto por un sistema inalámbrico adherido a un

¹⁰ INFORMADOR.MX, Le dan voz al lenguaje de señas con tecnología de Microsoft, tomado de: <http://www.informador.com.mx/tecnologia/2012/366529/6/le-dan-voz-al-lenguaje-de-senas-con-tecnologia-de-microsoft.htm>, Fecha de la consulta 2012.

guante, el cual posee un conjunto de sensores que capturan una serie de señales generadas por los movimientos gestuales de la mano, y un modelo por adaptación basado en los principios de la computación neuronal, el cual permite su reconocimiento en términos de un lenguaje dactilológico en particular. Los resultados arrojados por el sistema integrado mostraron gran efectividad en el reconocimiento de las vocales que conforman el lenguaje dactilológico en español, esto gracias a la capacidad que posee el modelo de asociar un conjunto de señales de entrada, con un movimiento dactilológico en particular.¹¹

Aparte de esto, en nuestro país, un ingeniero de sistemas colombiano, (Leal, 2012), desarrolló un traductor en internet que convierte el idioma español en el lenguaje de señas, este se trata, según sus palabras, de un "sistema unidireccional", es decir, del oyente la persona con discapacidad auditiva, que "hace un análisis gramatical" y utiliza "unos componentes de inteligencia artificial para encontrar una secuencia de imágenes" en el lenguaje de señas que corresponda a las palabras que previamente han sido digitadas¹².

A diferencia de los estudios nacionales presentados anteriormente, dos adolescentes sordomudas, estudiantes de la Institución Educativa Juan N Cadavid de Itagüí, expusieron en lenguaje de señas un dispositivo electrónico para mejorar la seguridad en las calles de las personas con discapacidad auditiva. El proyecto se exhibió con 130 investigaciones más, hechas por niños y jóvenes de Medellín, el Área Metropolitana, el Oriente y Urabá, en la Feria de la Ciencia la Tecnología y la innovación organizada por la Alcaldía de Medellín, EPM y Explora.

Debido a la alta accidentalidad que se presenta en las calles, (Isamar Cartagena & Katherine Fernández, 2013) diseñaron y construyeron este dispositivo llamado Vibrasor

¹¹ Escuela de Ingeniería de Antioquia (EIA). Envigado, Colombia, tomado de: (Revista Ingeniería Biomédica ISSN 1909-9762 / Volumen 7 / Número 13 / Enero-Junio de 2013 / pp.18-30. Fecha de consulta 2013.

¹² Semana. Colombiano crea traductor de lenguaje de señas para sordos. Tomado de: <http://www.semana.com/vida-moderna/articulo/colombiano-crea-traductor-lenguaje-senas-para-sordos/353935-3>. Fecha de consulta 2014

(palabra derivada de vibrador para sordos), que emite una señal vibratoria y luminosa para alertar a las personas con capacidad auditiva reducida.¹³

A partir del paso del tiempo son más las expectativas y con ello mayor el ingenio de los desarrolladores, Por ejemplo en Bogotá. (EFE). La fundación colombiana "Hablando con Julis", crea un software que permite a personas con discapacidad comunicarse sin intérprete¹⁴.

El programa permite a personas sordas, con síndrome de Down, afasia, autismo, parálisis cerebral o que por accidente o enfermedad perdieron la facultad para comunicarse construir frases o palabras para hacerse entender fácilmente.

Este software desarrollado en Colombia, combina alrededor de 44.000 palabras en español con imágenes, sonidos y videos. (Galindo, 2014).

¹³ Parque explora Medellín, Vibrasor, proyecto para sordos, tomado de:
<http://www.parqueexplora.org/educacion>, fecha de la consulta 2012.

¹⁴ software permite a los sordos comunicarse sin intérprete, tomado de:
<http://elcomercio.pe/tecnologia/inventos/software-permite-sordos-comunicarse-sin-interprete-noticia>, fecha de consulta 2014.

2.4. Justificación

El Informe mundial sobre la discapacidad (Organización Mundial de la Salud, 2011), propone medidas para todas las partes interesadas –incluidos los gobiernos, las organizaciones de la sociedad civil y las organizaciones de personas con discapacidad– para crear entornos favorables, promover la rehabilitación y los servicios de apoyo, asegurar una adecuada protección social, crear políticas y programas inclusivos, y aplicar normas y legislaciones, nuevas o existentes, en beneficio de las personas con discapacidad y la comunidad en general. Las personas con discapacidad deberán ocupar un lugar central en esos esfuerzos. Prueba de ello el informe de la BCFDD (2013) plantea la importancia de brindar un mayor acceso a las TIC para las personas con discapacidad, al mostrar cómo estas tecnologías pueden constituirse en un medio que les permita superar las limitantes de la discapacidad en el mundo laboral y desempeñarse eficientemente en la vida cotidiana¹⁵.

El Plan Nacional de Desarrollo del gobierno colombiano denominado “Prosperidad para Todos” 2010 – 2014 (Ministerio de salud y protección social, 2013) plantea en su capítulo IV. Igualdad de oportunidades para la prosperidad social, la necesidad de consolidar en el marco del sistema de protección social, todas las acciones conducentes a garantizar la inclusión social de las Personas con Discapacidad y sus familias

Durante los últimos años se ha tratado por medio de diferentes alternativas de ayudar a las personas con discapacidad auditiva para puedan comunicarse con el resto de la sociedad, es por ello que se han desarrollado diversos tipos de herramientas, entre las cuales se encuentra una tecnología denominada como estimuladores vibrotáctiles, los cuales son capaces de percibir la vibraciones emitidas por los diferentes tipos de sonidos sirviéndole como apoyo a los discapacitados para realizar una lectura un poco más acertada de los labios de una persona hablante; Los amplificadores de señal acústica, que

¹⁵ OMS, tomado de: www.who.int/disabilities/world_report/2011/summary_es.pdf, fecha de consulta 2011

es una manera de ayudar a las personas que no tiene pérdida total de la audición para que puedan seguir comunicándose con las demás personas; incluso aplicaciones móviles que tratan de alguna manera de ayudar un poco a minimizar esta problemática realizando una conversión del audio a texto permitiendo a los discapacitados entender de alguna forma lo que las personas están diciendo, esto asumiendo que el discapacitado en cuestión sepa leer, entre otras herramientas, pero no es solo esto, también se han estipulado leyes que apoyan a la comunidad sordo muda.

Con el desarrollo e implementación de un sistema de interpretación y traducción de gestos asociados a preguntas, necesidades y saludos básicos del lenguaje de señas colombiano, se busca contribuir de manera significativa a mejorar las relaciones interpersonales de las personas que padecen de discapacidades del oído y del habla (sordomudos) en todos los aspectos que para nosotros “personas del común” pueden resultar muy básicas.

3. MARCO TEÓRICO Y MARCO CONCEPTUAL

3.1. Marco Teórico

El reconocimiento, manipulación y virtualización de las imágenes a través de un sistema informático es un proceso muy complejo que debe considerar una gran cantidad de variables y aspectos muy importantes para su funcionalidad, a continuación se presentan las teorías, métodos y tecnologías necesarias para llevar a cabo el reconocimiento del lenguaje de señas colombiano.

3.1.2. Visión artificial

La visión artificial o visión es una disciplina científica que incluye métodos para adquirir, procesar, analizar y comprender las imágenes del mundo real con el fin de producir información numérica o simbólica para que puedan ser tratados por un computador. La característica principal de esta disciplina es permitir que un computador pueda reconocer una imagen o bien sea una secuencia de imágenes, permitiéndole capturar de esta una serie de parámetros numéricos, físicos y geométricos, ente otros, para luego utilizar estos parámetros como sea necesario para lograr el desarrollo de un software¹⁶.

3.1.3. Algoritmo de reconocimiento Adaboost

Esta metodología fue presentada originalmente por (Viola & Jones, 2001). En este se demuestra como a partir de características muy generales basadas en el cambio de intensidad se podía desarrollar un detector inteligente muy robusto (Rama & Tarrés, 2010)¹⁷. En términos generales este algoritmo me permite realizar operación con la imagen obteniendo los valores necesarios para su procesamiento, esto se realiza durante el entrenamiento donde las imágenes son convertidas a escala de grises con un tamaño por pixel definido para posteriormente extraer los datos¹⁸ como se mencionaba anteriormente este algoritmo sigue una secuencia estructurada de pasos descritos a continuación:

¹⁶ Vision artificial, tomado de: es.wikipedia.org/wiki/Visión_artificial, fecha de consulta 2014

¹⁷ Antonio Rama, Francesc Tarrés, nuevo método para la detección de caras basado en Integrales Difusas (Universitat Politècnica de Catalunya, Barcelona, Spain), fecha de consulta 2010

¹⁸ PAUL Viola, MICHAEL Jones. Robust Real- Time Object Detection. 2001

Pasos	Descripción
1.	Al ingresar las imágenes, estas deben estar pre-procesadas, es decir convertidas a escala de grises y redimensionada, para luego calcular las características de la misma.
2.	Se inician los pesos que se denomina como la recolección de los daos
3.	Se determina la cantidad de hipótesis T en el inicio del bucle
4.	Se normalizan los procesos para cada característica de la imagen evaluando los posibles errores
5.	Los procesos se pasan por un clasificador débil donde se evalúan valores característicos de la imagen
6.	Se escoge el clasificador con el menor error

Tabla 1. Pasos del algoritmo Adaboost¹⁹.

3.1.4. Interfaz de usuario natural (NUI)

este medio se cataloga como un tipo de interfaz de usuario que permite la interacción entre los humanos y los dispositivos, donde los disparadores de una acción están dados por agentes naturales como hablar, moverse, tener contacto visual con la aplicación, permitiendo al usuario clickear, seleccionar, ingresar, editar, etc.

Las interfaces de lenguaje natural componen un área activa de estudio en el campo del procesamiento del lenguaje natural y la lingüística computacional. Una intuitiva interfaz de lenguaje natural en general es uno de los grandes objetivos activos de la Web Semántica²⁰.

¹⁹ PAUL Viola, MICHAEL Jones. Robust Real- Time Object Detection. 2001

²⁰ Interfaz de usuario de lenguaje natural, tomado de:
es.wikipedia.org/wiki/Interfaz_de_usuario_de_lenguaje_natural#Procesado_y_reconocimiento_del_Lenguaje_Natural, fecha de consulta 2014

3.1.5. Kinect Studio v2.0

Kinect Studio hace de la codificación para aplicaciones de Windows un proceso mucho más fácil, además de eso incluye la librería más actual proporcionada, Kinect para Windows SDK 2.0, lo que permite a un desarrollador registrar todos los datos que pertinentes dentro del contexto de desarrollo de una aplicación a través de un sensor de Kinect v2.0. Esto significa que se pueden capturar los datos en una serie de movimientos del cuerpo o gestos específicos y luego utilizar esos datos una y otra vez de para depurar o mejorar el código. En pleno proceso de desarrollo posteriormente a la captura de los datos el Kinect Studio cuenta con la característica de realizar pruebas cada vez que se modifique el código sin la necesidad de tener el sensor conectado al ordenador aprovechando la fidelidad de la imagen en color, los datos de profundidad, y las relaciones tridimensionales. Con estos datos subidos a su computadora portátil a la mano, puede expresar en un poco o mucho de Kinect Studio²¹.

²¹ Kinect studio le permite código sobre la marcha, tomado de:
<http://blogs.msdn.com/b/kinectforwindows/archive/2015/07/10/kinect-studio-lets-you-code-on-the-go.aspx>, fecha de consulta: 2015

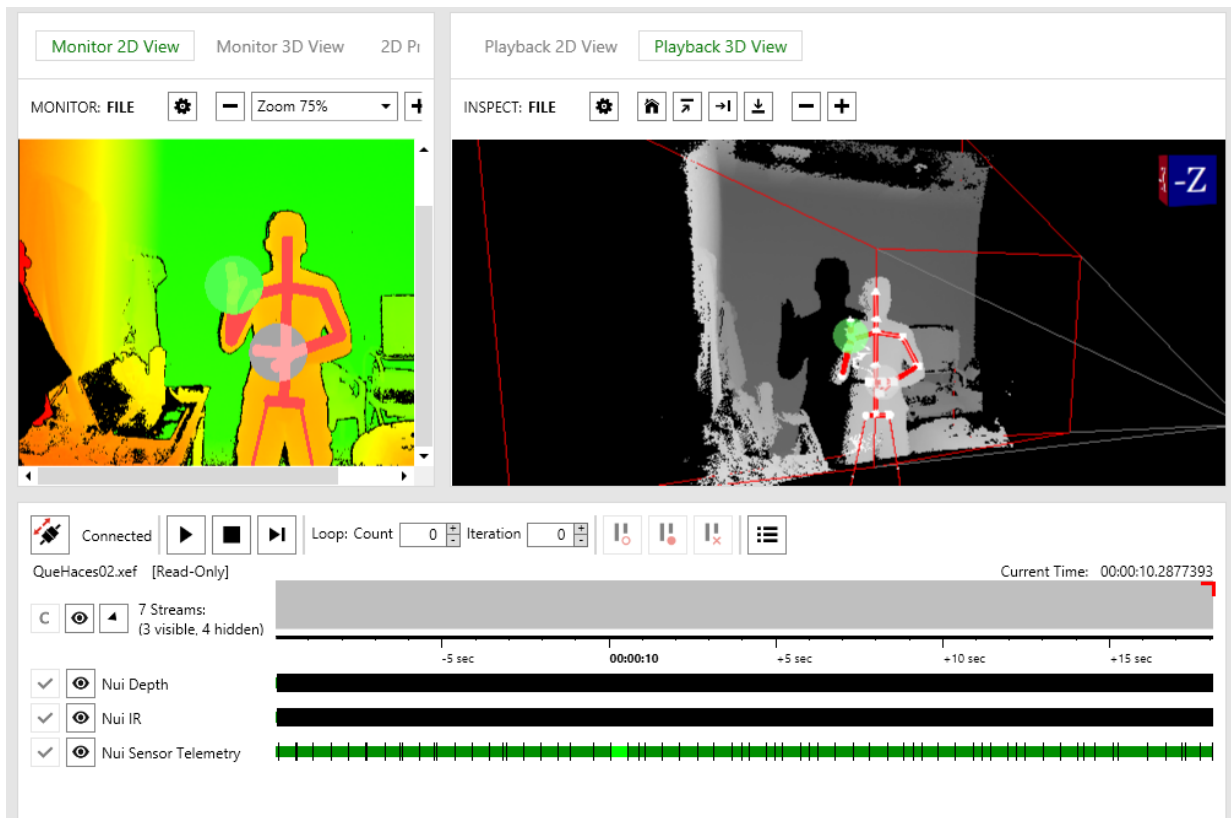


Figura 1. Grabando datos a través del Kinect Studio²².

3.1.6. Visual Gesture Builder

Gesto Constructor Visual (VGB) genera datos que las aplicaciones utilizan para realizar la detección de gestos en tiempo de ejecución. Incluso para los casos simples, detección gesto es una tarea difícil que puede requerir muchas líneas de código para obtener resultados fiables, teniendo en cuenta todos los diferentes usuarios y espacios que una aplicación podría encontrar. Mediante el uso de un modelo basado en datos, VGB cambia el énfasis de escribir código para la construcción de la detección gesto que es comprobable, repetible, configurable y con bases de datos. Este método proporciona un mejor reconocimiento de gestos y reduce el tiempo de desarrollo.

²² Figura tomada de: creación propia.

VGB utiliza una serie de tecnologías de detección. El usuario selecciona las tecnologías de detección que utilizan, es decir AdaBoostTrigger o RFRProgress y marcos de etiquetas en un clip relacionado con un gesto significativo, VGB construye una base de datos de gestos; con esta base de datos, una aplicación puede procesar la entrada del cuerpo de un usuario, por ejemplo, detectar un éxito o seguir un progreso²³.

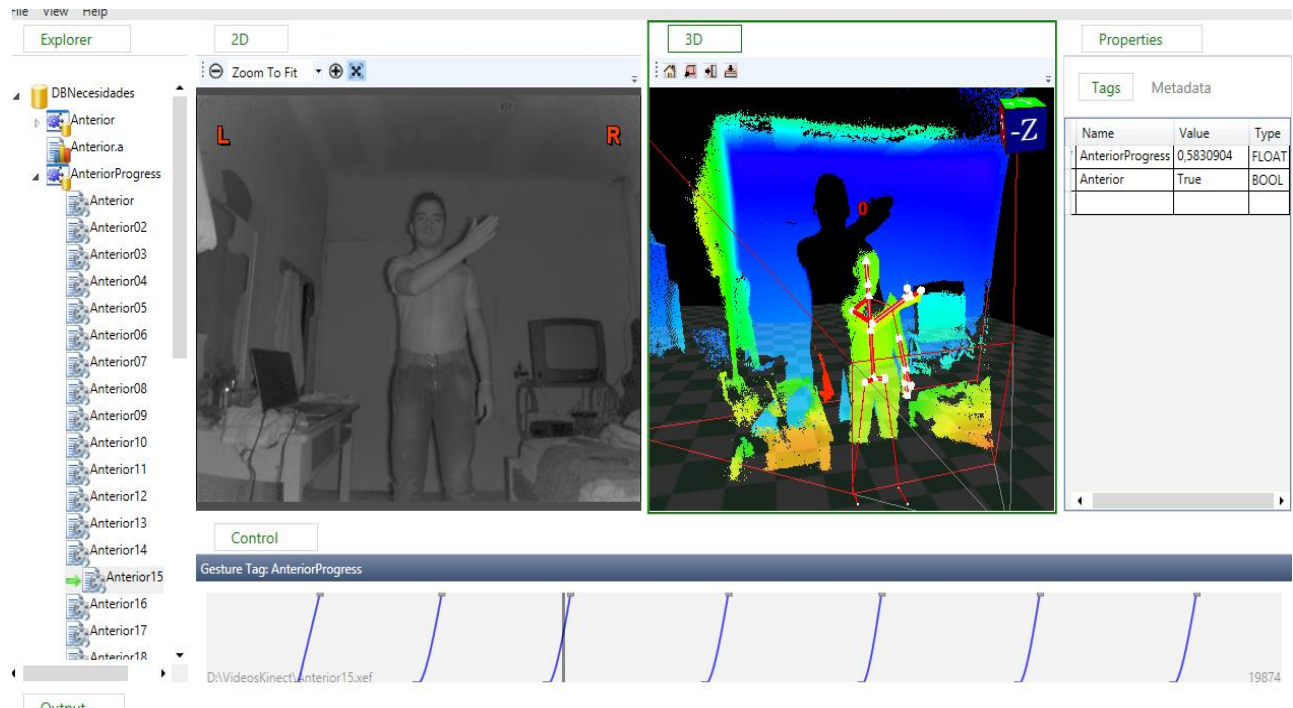


Figura 2. Alimentando la base de datos de gesto en el VGB²⁴.

²³ Gesto visual constructor, <https://msdn.microsoft.com/en-us/library/dn785529.aspx>, fecha de consulta 2015

²⁴ Figura tomada de: creación propia

3.1.6.1. AdaBoostTrigger

El AdaBoostTrigger es una tecnología de detección que produce un resultado binario o discreta. Utiliza un algoritmo adaptativo Impulso (AdaBoost) de aprendizaje automático (descrito en un apartado anterior), para determinar cuando un usuario realiza un cierto gesto.

Durante el tiempo de formación se validan etiquetas de entrada, valores booleanos, que marcan la realización de un gesto. Este etiquetado o marcado se utiliza para evaluar si es o no un gesto y determina el valor de confianza del evento²⁵.

3.1.6.2. RFRProgress

El RFRProgress es una tecnología de detección que produce un resultado análogo o continuo. Se utiliza el Random Bosque de regresión (RFR) algoritmo de aprendizaje automático para determinar el progreso de un gesto realizado por un usuario. Se utilizan modelos de marcado que están representados por las señales analógicas que se producen durante un gesto.

Existe una condición para que un gesto continuo se active, este solo se activa cuando se detecta un gesto discreto (AdaBoostTrigger) previamente²⁶.

3.1.7. Percepción de profundidad

A través de la visión binocular somos capaces de interactuar en un mundo tridimensional, pues podemos apreciar las distancias y el volumen de los objetos en el entorno que nos rodea.

²⁵ Adaboosttrigger, tomado de: msdn.microsoft.com/en-us/library/dn785522.aspx, fecha de consulta: 2014

²⁶ RFRProgress, tomado de: msdn.microsoft.com/en-us/library/dn785524.aspx, fecha de la consulta 2014

Nuestros ojos, debido a su separación, obtienen dos imágenes con pequeñas diferencias entre ellas, conocidas como disparidad. Nuestro cerebro procesa las diferencias entre ambas imágenes y las interpreta de tal forma que percibimos la sensación de profundidad, lejanía o cercanía de los objetos que nos rodean. Este proceso se denomina estereopsis²⁷.

3.1.8. Mapa profundidad Kinect

El Kinect es un dispositivo capaz de medir la distancia que existe entre la cámara que percibe los rayos infrarrojos que el dispositivo emite y su entorno lo que crea una nube de puntos (figura 11) que son capturados y procesados internamente por el dispositivo Kinect produciendo una matriz de valores fraccionarios que se conoce como mapa de profundidad.

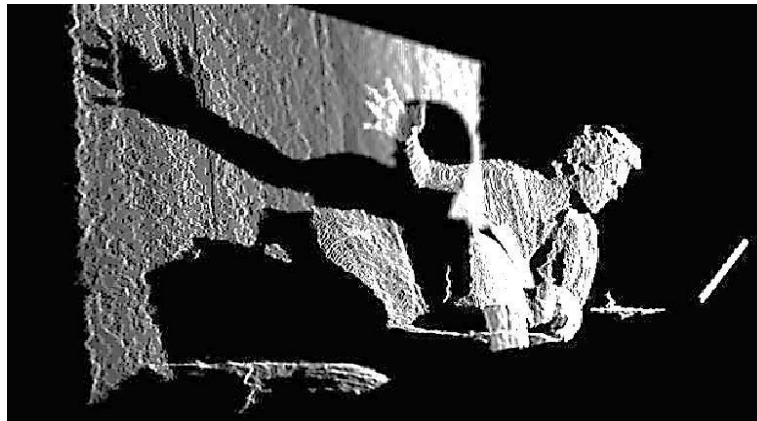


Figura 3. Mapa de profundidad.²⁸

²⁷ Ogle, Kenneth N., *Researches in binocular vision*, Oxford, England: W. B. Saunders, Inglaterra, 1950

²⁸ Mapas de profundidad, tomado de: http://ocw.unia.es/ciencias-tecnologicas/tecnologia-del-ocio/materiales-basicos-folder/html/B1_UD03/mapas_de_profundidad.html/skinless_view, 2012

Esta matriz no tiene una representación visual directa al no pertenecer a los modelos de color visibles por el ojo humano, por lo que para poder tener una representación visual en pseudocolor, las distancias son transformadas a un modelo de color generalmente en escala de grises como se puede observar en la figura 4.

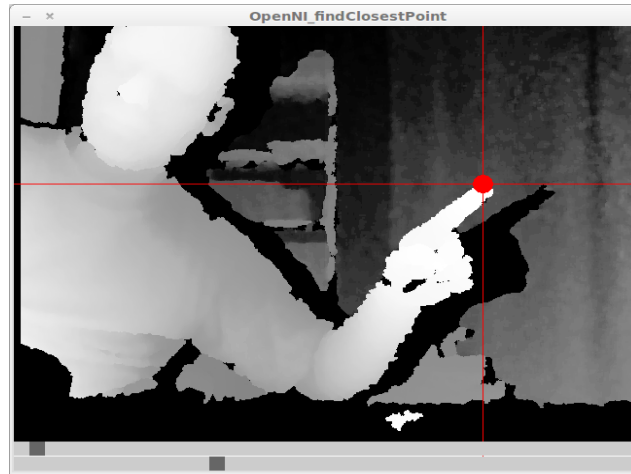


Figura 4. Transformación a escala de grises.²⁹

3.1.8.1. Skeletal Tracking.

“Es la funcionalidad estrella del sensor Kinect, significa seguimiento del esqueleto y se basa en un algoritmo que logra identificar partes del cuerpo de las personas que están en el campo de visión del sensor” (esmsdn, 2011). Por medio de este algoritmo podemos obtener puntos que hacen referencia a las partes del cuerpo de una persona y hacer un seguimiento de estos identificando gestos y/o movimientos.³⁰

²⁹ Jorge ivan meza, tomado de <http://blog.jorgeivanmeza.com/tag/kinect/>, 2013.

³⁰ Kigo el robot clasificador de residuos, tomado de, <http://www.lsedkigo.blogspot.com/2012/05/integracion-de-kinect-parte-i.html>, 2011

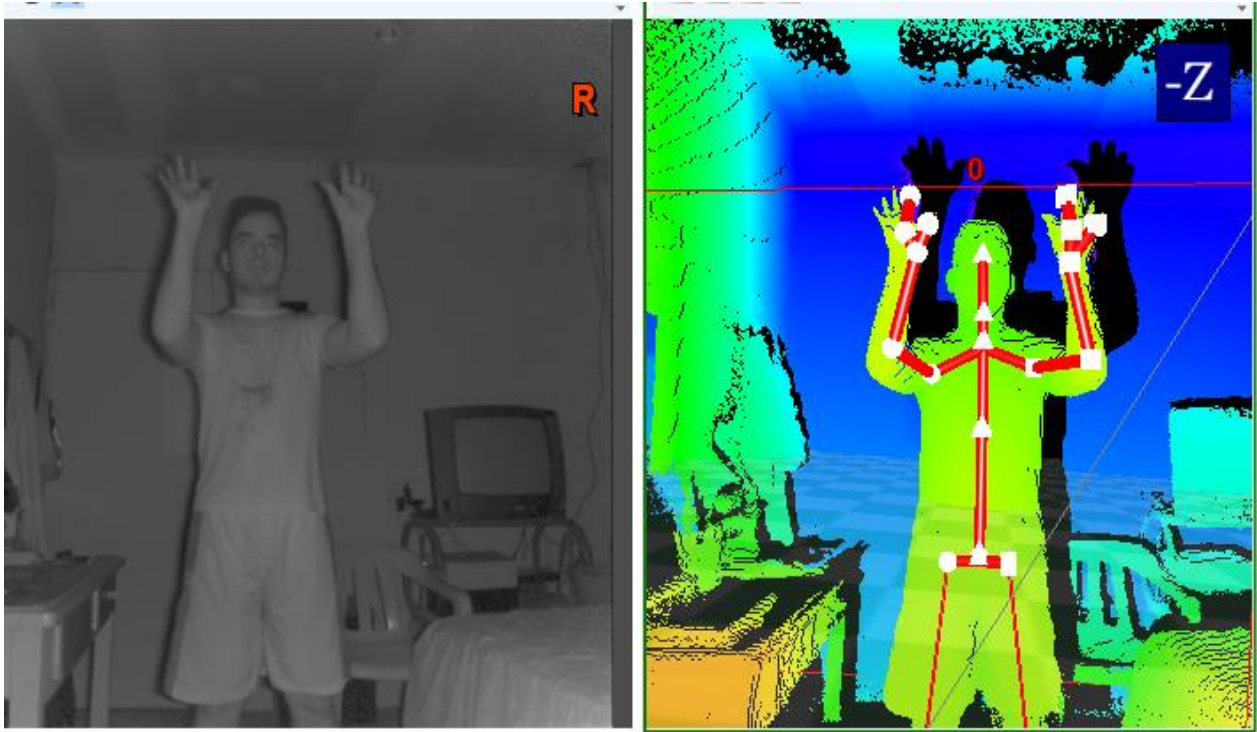


Figura 5. Skeletal Tracking³¹

3.1.8.2. Kinect para Windows SDK 2.0

El Kinect para Windows Software Development Kit (SDK) 2.0 permite a los desarrolladores crear aplicaciones que soportan el gesto y reconocimiento de voz, utilizando tecnología de sensor Kinect en equipos con Windows 8, Windows 8.1, y Standard 8 Windows Embedded³².

El kit de desarrollo de software de Windows (SDK) v2.0 le permite al desarrollador crear aplicaciones y experiencias comerciales o Windows Store que soportan gestos y

³¹ Figura tomada de: creación propia.

³² Kinect for windows sdk 2.0, tomado de: www.microsoft.com/en-us/download/details.aspx?id=44561, fecha de consulta 2014

reconocimiento de voz mediante el uso de C ++, C #, Visual Basic, o cualquier otro lenguaje .NET o proyección Windows Store³³.

3.1.9. Windows Presentation Foundation (WPF).

Windows Presentation Foundation (WPF) permite el desarrollo de interfaces en Windows con un lenguaje de programación .NET. Ofrece una amplia infraestructura y potencia grafica con la que es posible desarrollar aplicaciones visualmente atractivas. Permite integrar elementos de animación, videos, imágenes y audio de manera sencilla, cosa que viene bien para el desarrollo de este proyecto.

WPF utiliza el lenguaje XAML. La ventaja específica que XAML lleva a WPF es que XAML es un lenguaje completamente declarativo. En un lenguaje de programación declarativa, el desarrollador (o diseñador) describe el comportamiento y la integración de los componentes sin utilizar programación procedural. Aunque es raro que una aplicación completa se construya totalmente en XAML, la introducción de XAML permite a los diseñadores de aplicaciones contribuir más eficazmente al ciclo de desarrollo de aplicaciones (Wikipedia, 2013).

³³ Herramientas y recursos de Kinect, tomado de: dev.windows.com/en-us/kinect/tools, 2014

3.2. Marco conceptual

Dentro del transcurso del desarrollo de este proyecto se maneja mucha información respecto a teorías, estudios, publicaciones y desarrollos realizados en base a la problemática sobre la cual se basa este trabajo, por ello se hace necesaria la definición de algunos de los conceptos más importantes contemplados dentro del cuerpo de dicho trabajo.

3.2.1. Discapacidad

Discapacidad es un término general que abarca las deficiencias, las limitaciones de la actividad y las restricciones de la participación. Las deficiencias son problemas que afectan a una estructura o función corporal; las limitaciones de la actividad son dificultades para ejecutar acciones o tareas, y las restricciones de la participación son problemas para participar en situaciones vitales.

3.2.2. Sordomudo

Es un término que designa a aquellas personas que no han desarrollado o han perdido la capacidad auditiva y vocal al mismo tiempo. No confundir con personas sordas, con discapacidad auditiva, sordo ciegas, o mudas

3.2.3. Visión artificial

La visión artificial, también conocida como visión por computador o visión técnica, es un sub campo de la inteligencia artificial. El propósito de la visión artificial es programar un computador para que "entienda" una escena o las características de una imagen³⁴.

³⁴ Wikipedia enciclopedia libre, tomado de http://es.wikipedia.org/wiki/Vision_artificial, 2013

3.2.4. Interface natural de Usuario

Interface natural es manejar el sistema y/o aplicación a través de tu cuerpo, gestos, voz, etc., la ventaja es que no necesitas dispositivos o periféricos de entrada para dar órdenes al sistema como tal.

3.2.5. NET

.NET es un framework de Microsoft que independientemente de plataformas virtuales o del hardware facilita el desarrollo de aplicaciones y además busca interconectar servicios para ofrecer mucha mejor experiencia a los desarrolladores.

3.2.6. Lenguaje XAML

XAML es un lenguaje declarativo de marcado Tal y como se aplica en el modelo de programación .NET Framework, XAML simplifica la creación de la interfaz de usuario para una aplicación .NET Framework. Se pueden crear elementos visibles de la interfaz de usuario en el marcado XAML declarativo y, a continuación, separar la definición de la interfaz de usuario de la lógica en tiempo de ejecución mediante archivos de código subyacente, que se unen al marcado mediante definiciones de clases parciales³⁵.

3.2.7. Programación declarativa

La Programación Declarativa, en contraposición a la programación imperativa es un paradigma de programación que está basado en el desarrollo de programas especificando

³⁵ Microsoft, tomado de [http://msdn.microsoft.com/es-co/library/ms752059\(v=vs.110\).aspx#what_is_xaml](http://msdn.microsoft.com/es-co/library/ms752059(v=vs.110).aspx#what_is_xaml), 2012

o "declarando" un conjunto de condiciones, proposiciones, afirmaciones, restricciones, ecuaciones o transformaciones que describen el problema y detallan su solución³⁶.

3.2.8. Sensor Kinect

Kinect (originalmente conocido por el nombre en clave <<Project Natal>>), es <<un controlador de juego libre y entretenimiento>> creado por Alex Kipman, desarrollado por Microsoft para la video consola Xbox 360, y desde Junio del 2011 para PC a través de Windows 7 y Windows 8. Kinect permite a los usuarios controlar e interactuar con la consola sin necesidad de tener contacto físico con un controlador de videojuegos tradicional, mediante una interface natural de usuarios que reconoce gestos, comandos de voz y objetos e imágenes.

En cuanto al dispositivo en sí³⁷, consiste en una barra horizontal de aproximadamente 23 centímetros conectada a una pequeña base circular con un eje de articulación de tipo rótula. El dispositivo final está diseñado para ser colocado longitudinalmente por encima o por debajo de la pantalla de visualización, ya sea un monitor o un televisor convencional.

El dispositivo cuenta, a nivel de hardware, con una cámara RGB, un sensor de profundidad, un micrófono de múltiples matrices y un procesador propio personalizado que se encarga de ejecutar el software patentado, que proporciona captura de movimiento de todo el cuerpo en 3D en tiempo real, reconocimiento facial y capacidades de reconocimiento de voz. El micrófono de matrices del sensor puede permitir llevar a cabo la localización de la fuente acústica y la supresión del ruido ambiente.

³⁶ Wikipedia enciclopedia libre, tomado de http://es.wikipedia.org/wiki/Programación_declarativa 2011

³⁷ Kinect Documentation, tomado de <http://www.microsoft.com/enus/kinectforwindows/develop/resources.aspx>, 2012.



Figura 6. Sensor Kinect v2.0³⁸

"El dispositivo Kinect consta de tres partes: una luz infrarroja, la luz invisible que ilumina a los objetos que encuentra dentro de su campo de visión; un Sensor CMOS o Active Pixel Sensor (APS), sensor que detecta la luz basado en tecnología CMOS y que es usado en cámaras DSLR, que detecta las diferentes distancias (intensidad) que recorre la luz; y el chip PrimeSense 1080 SoC que mediante un algoritmo muy sofisticado traduce la información recibida de la luz en un mapa de profundidad. Todo en tiempo real."³⁹

El sensor IR ilumina todos los objetos que están dentro de su campo de visión. Esta luz rebota en los cuerpos y viaja de nuevo hasta el Kinect donde el receptor captura la información de la distancia recorrida por la luz por cada pixel y genera una malla cartesiana asignado a cada pixel una profundidad.

Además también recibe información a través de su cámara RGB, por lo que también disponemos por pixel, de la información de color.

En cuanto a la utilización en PC del sensor, fue la propia Microsoft quien dejó de forma gratuita y a disposición de todo el mundo los drivers para el control de Kinect v2.0 para Windows.

³⁸ Taringa, tomado de <http://www.mundogamers.com/noticia-y-si-xbox-one-se-convierte-en-el-steam-consolero.2632.html>, 2013

³⁹ Pisito en Madrid, tomado de: <http://www.pisitoenmadrid.com/blog/2011/02/experiencia-kinect/>, 2011

La comunidad “Kinesis” nos provee con SDK, el cual es una librería JavaScript, que nos permite desarrollar aplicaciones de escritorio o web basadas en gestos gracias al Kinect, esta funciona con Xbox y Windows Kinect.

3.2.8.1. Componentes del Kinect



Figura 7. Componentes del Kinect v2.0.

3.2.8.2. Sensor de profundidad.

El sensor de profundidad es un proyector de infrarrojos con un sensor de tipo CMOS monocromo que permite a Kinect ver la habitación en la que se encuentre en 3D bajo cualquier condición de luz ambiental. El rango de detección de la profundidad del sensor es ajustable gracias al software interno de Kinect que es capaz de calibrar automáticamente el sensor, basado en la jugabilidad y en el ambiente físico del jugador, tal como puede ser la presencia de otros objetos, como por ejemplo sillas y sofás.

3.2.8.3. Cámara RGB.

La cámara RGB (rojo, verde, azul) de la Kinect. Es del tipo CMOS, trabaja por defecto de **1920 x 1080 Full HD**.

3.2.8.4. Micrófono multi-matriz.

En la parte delantera del Kinect hay una serie de micrófonos que se usan para el reconocimiento de órdenes y charla.

3.2.9. Visual Studio 2013

Microsoft Visual Studio es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta múltiples lenguajes de programación tales como C++, C#, Visual Basic .NET, F#, Java, Phyton, Ruby, php; al igual que entornos de desarrollo web como ASP.NET MVC, Django, etc.

Visual Studio permite a los desarrolladores crear aplicaciones, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET (a partir de la versión .NET 2002). Así se pueden crear aplicaciones que se comuniquen entre estaciones de trabajo, páginas web, dispositivos móviles, dispositivos embebidos, consolas (la Xbox 360 y Xboxone), etc (Wikipedia, 2013).

3.2.10. Lenguaje de programación C#.

C# (pronunciado si Sharp en inglés), es un lenguaje de programación orientado a objetos desarrollado y estandarizado por Microsoft como parte de su plataforma .NET. Su sintaxis básica deriva de C/C++ y utiliza el modelo de objetos de plataforma .NET, similar al de Java, aunque incluye mejoras derivadas de otros lenguajes.

Aunque C# forma parte de la plataforma .NET, esta es una API, mientras que C# es un lenguaje de programación independiente diseñado para generar programas sobre dicha plataforma. Ya existe un compilador implementado que provee el marco Mono – DotGNU, el cual genera programas para distintas plataformas como Windows, Unix, Android, IOS, Windows Phone, Mac OS y GNU Linux (Wikipedia, 2012).

4. METODOLOGÍA

4.1. Tipo de investigación

Este proyecto se realizó bajo un tipo de investigación descriptiva, puesto que para iniciar con el desarrollo del mismo, primero se llevó a cabo un análisis exhaustivo de la información pertinente a este, recopilado de artículos, revistas, trabajos realizados y demás, para posteriormente determinar puntualmente los alcances, limitaciones y resultados que se pueden llegar a presentar. Y así partiendo de esta determinación de factores influyentes determinar qué tipo de tecnología podría ofrecer mejores resultados para el desarrollo del proyecto.

4.2. Línea de investigación

El proyecto diseño e implementación de un sistema de interpretación y traducción para el vocabulario básico del lenguaje de señas colombiano. Se enmarca dentro de las líneas de investigación de la ingeniería de software, ya que se hace necesario el diseño y desarrollo de un software, que permita la asociación gesto - espacial de estímulos realizados, a través de una interfaz natural de usuario, estructurada bajo un algoritmo de reconocimiento de gestos.

4.3. Población Objetivo

Este proyecto pretende ser desarrollado y aplicado a todas las personas de la comunidad sordo – muda Colombiana que manejen el lenguaje señas del respectivo país; Y basandome en las esclarecedoras cifras aportadas por el (DANE , 2005) , ya expuestas anteriormente, y obviando según este mismo estudio que el porcentaje de población en discapacidad, es del 17.3%, representados en 455.718 individuos con limitación

permanente para oír.⁴⁰, se especifica a que cantida y que tipo especifico de personas esta dirigido este proyecto.

En la siguiente figura se enmarca la poblacion hacia la cual va dirigido el objetivo principal del proyecto en color rojo.



Figura 8. Población Objetivo.⁴¹

4.4. Faces del proyecto

El desarrollo de un proyecto, implica casi siempre una serie de procesos estructurados que deben ser ejecutados secuencialmente. Por ello es posible establecer una serie de fases o etapas de trabajo, relativamente diferenciadas entre sí, que marcan la evolución del proyecto desde el planteamiento del problema hasta su solución. Cada una de estas etapas juega un papel crucial para el desarrollo del proyecto.

⁴⁰ Oir Audiologia, tomado de <http://oiraudiologia.blogspot.com/2014/03/situacion-de-dicapacidad-auditiva-en.html>, sitio web 2012

⁴¹ Figura tomada de: creación propia

En otras palabras, ninguna de las fases de desarrollo puede ser obviada u omitida sin poner en riesgo la integridad del proyecto. A continuación se muestra gráficamente la secuencia que siguen las fases de desarrollo.

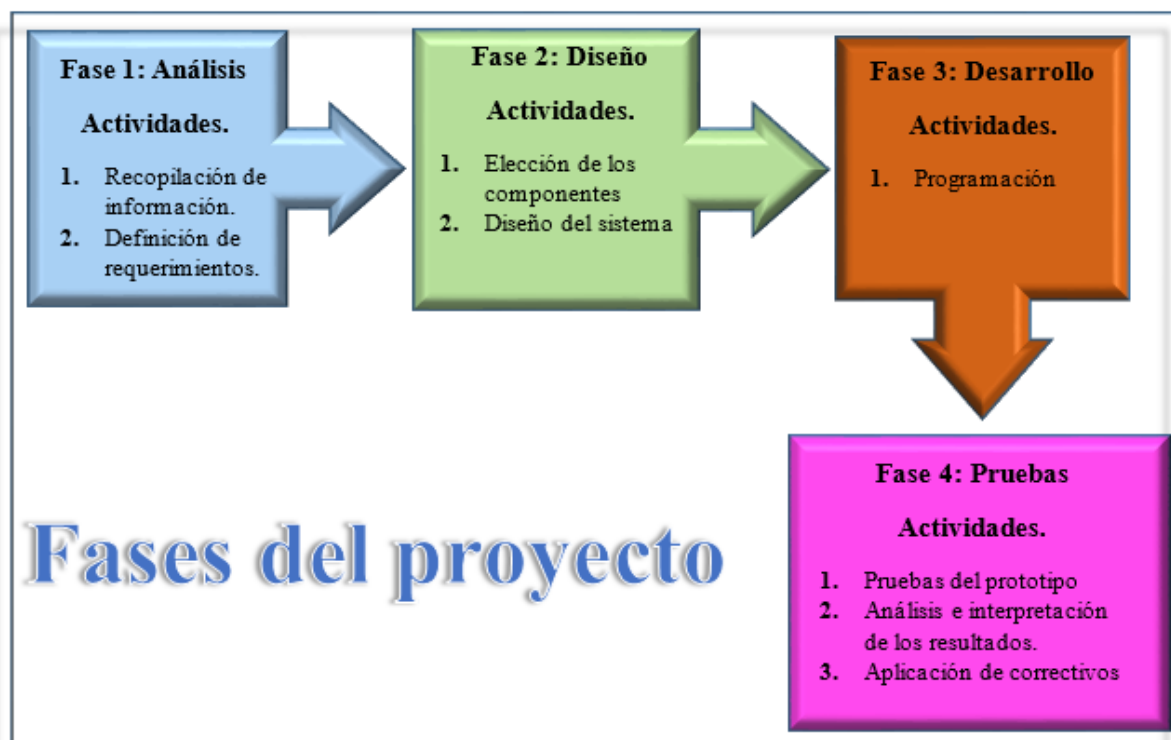


Figura 9. Fases del proyecto⁴².

4.4.1. Fase 1: Análisis

En esta fase se realizó un estudio sobre el estado actual de la problemática, analizando datos estadísticos, estudios y trabajos realizados, acerca de cualquier tipo de avance que se haya desarrollado con el propósito de solventar esta problemática tanto a nivel internacional como nacional, en los cuales se evidencia la situación actual de las personas que padecen algún tipo de discapacidad y más específicamente las personas con discapacidad auditiva y del habla.

⁴² Figura tomada de: creación propia

Una vez recolectada toda la documentación relevante, fue necesario realizar la definición de todas las variables, y sus posibles interrelaciones, además la formulación de los requerimientos (funcionales y no funcionales) y también un análisis de los desarrollos hechos anteriormente y en la actualidad en torno a esta problemática.

Para esta fase se siguen las siguientes actividades:

4.4.1.1. Actividad 1: Recopilación de información

En primera instancia se indago al dentro de múltiples fuentes que contienen datos estadísticos como los informes presentados por la (Organización Mundial de la Salud, 2015) y el (Ministerio de salud y protección social, 2013), en estos apartados se describe de forma muy detallada la situación que padecen las personas con discapacidad del oído y del habla (a nivel mundial y nacional). Datos que revelan los altos índices de complejidad que rodean a este problema.

Luego se consultó a través de diversas fuentes de información como revistas especializadas, resultados de encuestas, estudios realizados, informes nacionales y en la web, buscando proyectos, o aplicaciones desarrollados que contribuyan mejorar la calidad de vida de las personas con discapacidades auditivas y del habla.

Seguido a encontrar las fuentes de información, proseguía la selección de los documentos que si representaban una base o un antecedente, es decir, luego de haber filtrado la información que si iba acorde con el estudio que se plantea en este proyecto, se procedió a seleccionar de cada uno de los datos más relevantes que fueran de la mano con el tipo de proyecto que se quería desarrollar, también las tecnologías y el nivel de aportes o logros que estos estudios generaron para contribuir de forma significativa en la inclusión de los discapacitados como seres independientes en la sociedad.

4.4.1.2. Actividad 2: Definición de requerimientos

Dentro de la segunda fase se procede a realizar la definición y el establecimiento de los requerimientos (funcionales y no funcionales) a los cuales debe estar sujeto el sistema para tener una estructura igual a la que fue planteada en el diseño del proyecto, esto como referente a la elaboración del mejor software posible según las especificaciones planteadas.

Los requerimientos se formularon de una forma precisa y acertada, con el propósito de que sean interpretados correctamente para prevenir que a la hora de ser instaurados y realizados a los largo del proceso de desarrollo del sistema se haga de manera errada.

4.4.2. Fase 2: Diseño

En base a la información categorizada en la fase anterior se determina que tecnología para el reconocimiento corporal va a ser implementada en el desarrollo del proyecto, teniendo en cuenta aspectos como la fidelidad y la calidad de los datos ofrecidos por dicha tecnología. Además de eso se elaboraron los diagramas UML (diagrama de actividades, de secuencia, casos de uso), en donde se muestran gráficamente las transiciones internas entre procesos que se dan en el sistema.

La fase de diseño contiene 2 actividades, la elección del sensor y el diseño del sistema.

4.4.2.1. Actividad 1: Elección del sensor

Luego de analizar detenidamente la información encontrada en torno a las tecnologías usadas en el desarrollo de los diferentes dispositivos y aplicaciones ya creadas que

ofrecen resultados similares a los que se esperan con la elaboración de este proyecto, se procede a seleccionar el dispositivo más adecuado para el desarrollo de este proyecto.

A partir de la selección del dispositivo a usar en el desarrollo del sistema, también se determinaron otros elementos necesarios para realizar las conexiones tales como adaptadores y cables especiales.

4.4.2.2. Actividad 2: Diseño del sistema

Dentro esta fase se dan por sentados por decirlo de esta forma los planos estructurales del sistema, ya que es aquí donde se imagina y se plasma visualmente como va estar constituido el sistema por ello en esta fase se realizaron los diagramas UML (diagrama de actividades, de secuencia, casos de uso), con los que se muestra gráficamente cómo se comporta el sistema a la hora de realizar las respectivas interacciones con el usuario y cada uno de los procesos asociados a la administración de la información y además de eso las decisiones tomadas según los diversos condicionales. Por último se esquematizo la organización y las relaciones entre los componentes representados en la arquitectura del sistema.

4.4.3. Fase 3: Desarrollo

Esta fase está directamente relacionada con la fase anterior, el diseño del sistema, puesto que a partir del diseño con sus respectivas consideraciones, atributos, excepciones y contemplaciones es que podemos pensar en iniciar el desarrollo y es precisamente por eso que esta fase hace énfasis en la estructuración y construcción sistema.

No obstante, este sistema se desarrolló basándose en los requerimientos, diagramas y la arquitectura del sistema definidos en apartados anteriores del proyecto. Aparte de eso es en esta fase donde se inició la elaboración del algoritmo que posteriormente se encargaría del reconocimiento de los gestos pertinentes al lenguaje de señas

Colombiano. Se implementaron los mecanismos de respuesta y validación de las lecturas provenientes del sensor.

Y finalmente se procedió a instalar los diferentes controladores del dispositivo, el equipo en el cual se iba a trabajar, para lo cual se realizó una conexión cableada a través de un adaptador especializado; La presente fase tiene una única actividad, la cual es la programación.

4.4.3.1. Programación

Luego de realizar satisfactoriamente las actividades inmediatamente anteriores, se inician los procesos relacionados con la programación del sistema. El software desarrollado se pensó y elaboró basándonos principalmente en los requerimientos determinados en la fase de análisis, al igual que teniendo muy presente la orientación hacia las especificaciones planteadas en el diseño. Se realizaron paso a paso los procesos definidos en los diagramas para obtener un óptimo funcionamiento del sistema.

Después de eso el siguiente paso era continuar hacia el desarrollo del software, encargado de interpretar las señales provenientes del usuario y captadas por el sensor. Se programan las funciones del dispositivo como la calibración de cada nuevo usuario, la validación para cada tipo de gesto, la identificación de los gestos y la respuesta generada como respuesta al estímulo generado por los gestos del usuario anteriormente mencionados. Luego se elaboró cada una de las partes que componen la interfaz de usuario.

4.4.4. Fase 4: Pruebas

Una vez concluido el proceso de la programación del sistema como tal se deben realizar pruebas para corroborar las variables y procesos funcionales del mismo, luego es prudente realizar un análisis de los diferentes diagramas para confirmar que la estructura del sistema está realizada acorde con estos, ya que estos diagramas son los que deben regir los procesos realizados por el sistema y de esta forma interpretar cada lectura del sensor así como las respuestas del sistema.

Las Pruebas están conformadas por 3 actividades descritas a continuación.

4.4.4.1. Actividad 1: Pruebas del sistema

En esta fase se procede a realizar las pruebas respectivas luego de la finalización del desarrollo del prototipo analizando y verificando el resultado de los procesos que lo componen dependiendo de cada una de las acciones tomadas por el usuario.

A partir de aquí se realizó un proceso de evaluación de cada una de las respuestas generadas por el sistema dependiendo de las acciones realizadas por el usuario con el propósito de determinar la veracidad de las respuestas asociadas a cada prueba o en su defecto determinar los posibles errores que se pudieran presentar.

Independientemente de las evaluaciones realizada para verificar el comportamiento del sistema, se realizaron más y más pruebas, ya que con la realización de una prueba cualquiera surgían casos como este ejemplo: “Funciono de esta manera, pero... ¿será que si pruebo de esta otra forma también funcionará?”, preguntas a las cuales había que darles una respuesta a través de la realización de una nueva prueba que respondiera a cada nuevo caso en específico.

4.4.4.2. Actividad 2: Análisis e interpretación de los resultados

Se hizo un análisis exhaustivo de todos los resultados arrojados por cada uno de los tipos de prueba que se realizaron en la fase anterior para determinar si dichos resultados hacían referencia o no a los requerimientos funcionales establecidos anteriormente para el óptimo funcionamiento del sistema y además de eso verificar si estos datos obtenidos de las pruebas correspondían a el tipo de datos que debía proporcionar el sensor en cada una de los diferente escenarios de prueba. Luego se tomaron las decisiones sobre los correctivos de que debían ser aplicados a las diversas fallas detectadas durante las pruebas.

4.4.4.3. Actividad 3: Aplicación de correctivos

Posteriormente a la identificación de las fallas que se presentaron durante las pruebas, se procedió a realizar cambios, a redefinir la estructura de la composición de los gestos, se pasó de trabajar con un tipo de gestos discretos, es decir, que responden a valores Booleanos (verdadero o falso), a trabajar con gestos de tipo continuo, es decir, que responden a valores más flexibles de tipo entero (porcentajes y números decimales); Además de esto también se procedió a subdividir la base de datos que existía en 3 diferentes bases de datos categorizando en cada una con un tipo específico de gestos, para así obtener un prototipo funcional, el cual será el resultado de las distintas correcciones hechas a partir del desarrollo del primer prototipo, al terminar con este arduo proceso de identificación y corrección de errores el sistema fue capaz de cumplir con los requerimientos y objetivos establecidos a lo largo del desarrollo del sistema, logrando una versión del sistema perfectamente funcional.

4.5. Metodología de desarrollo del producto

Este proyecto fue desarrollado bajo un concepto organizacional en cascada, el cual está conformado por una sucesión de eventos o fases intrínsecamente relacionados el uno en función del otro, cada uno contiene un conjunto de actividades indispensables, los cuales deberán cumplir con un tiempo definido entre su inicialización y su culminación. Además de eso antes de proseguir con el evento o fase siguiente es necesario finalizar el actual que se encuentra en progreso.

La representación gráfica del modelado de desarrollo en cascada se ilustra a continuación.

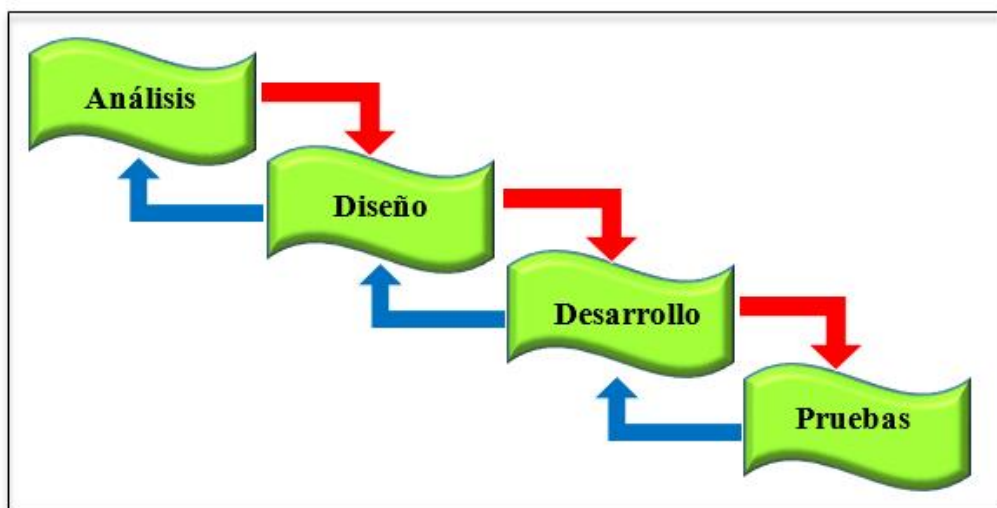


Figura 10. Modelo de desarrollo.⁴³

Las fases secuenciales que se deben seguir durante la ejecución de este modelo de desarrollo, se subdividen cada una en diferentes actividades, que deben ser cumplidas estructuralmente como lo muestra la gráfica del modelo, sin pasar de una actividad a otra, mientras no se termine la fase inmediatamente anterior.

⁴³ Figura tomada de: creación propia

Las cuatro etapas de este modelo de desarrollo, son específicamente las siguientes.

4.5.1. Análisis

Esta etapa es la más significativa e importante en lo que a planeación de procesos se refiere, ya que en esta es donde se define en primera instancia que es lo que se quiere hacer, como se va a hacer y con que se puede contar para hacerlo. Esta fase inicial se ramifica en dos actividades, la primera actividad es la recopilación de información, la cual está definida por la búsqueda de información referente a desarrollos de proyectos, artículos, revistas, publicaciones y cualquier tipo de estudio estadístico relacionados con el proyecto. La segunda actividad trata la definición de los requerimientos, en esta se especificaron las funciones del sistema y características específicas es las cuales se enmarca el desenvolvimiento del sistema y la estructura fundamental de su uso como tal.

4.5.2. Diseño

A partir de aquí se seleccionaron las herramientas de software y hardware necesarias en la construcción del software y por ende necesarios también para la interacción de su arquitectura y las interrelaciones que existen entre los ya mencionados hardware y software.

El diseño comprende la documentación estructural de los diagramas y los parámetros asociados al funcionamiento del sistema en esta actividad se describen los procesos que van a ser implementados dentro del sistema con sus respectivos elementos asociados e interrelaciones formadas entre sus diferentes atributos.

El comportamiento del interno del sistema se describe gráficamente en los diagramas UML conformados por diagramas (diagrama de actividades, de secuencia, casos de uso,

de clases), en estos se muestra el comportamiento del sistema a la hora de que el usuario interactúe con él.

4.5.3. Desarrollo

El desarrollo del sistema se implementó segmento a segmento, modelo que permitió que el sistema lograra cumplir con todos los requerimientos exigidos, de acuerdo a las especificaciones definidas en las respectivas etapas antes descritas.

En el proceso de la elaboración del sistema se utilizó un único dispositivo, el sensor de reconocimiento de gestos Kinect para determinar, los diferentes movimientos y gestos correspondientes a cada tipo de frase que en general forman parte del vocabulario básico de señas Colombiano, los factores correctos para la detección del movimiento y calibración de un cuerpo como tal, se obtienen mediante la programación, que se evalúan utilizando el Kinect, el cual permite mediante una serie algoritmos y métodos desarrollados, validar los diferentes estados posicionales a la hora de expresar palabras o frases pertenecientes a la lengua de señas de nuestro país.

4.5.4. Pruebas

Acto seguido a que el sistema fuera integrado por completo y culminado en términos de desarrollo, se realizaron las diferentes pruebas, de tal forma que se confirmara si el sistema cumplió con las especificaciones y definiciones del software, después del sometimiento a las diferentes pruebas y una vez ejecutadas positivamente se elaboró toda la documentación requerida. Instalado el sistema y puesto en funcionamiento, se corrigió los posibles errores, no detectados antes, tanto en el software como el hardware, se adicionaron posibles mejoras, de acuerdo a nuevos requerimiento.

5. DESARROLLO

5.1.Arquitectura del sistema

La arquitectura del sistema es una observación superficial y global de alto nivel que representa la forma en cómo se comunican los componentes. Es una perspectiva del sistema totalmente libre de algoritmo y programación es capaz de mostrar desde la estructura más interna hasta la más superficial, de igual forma también puede mostrar de qué manera se genera el flujo de datos y la organización de los procesos asociados al sistema.

Se agrupan para conformar subsistemas un terminal con Windows, un sensor Kinect y una serie de librerías y drivers como se evidencia en la estructuración de las diferentes capas que a su vez tienen otros elementos que conforman el sistema.

La siguiente figura representa la arquitectura seguida en el desarrollo del sistema.

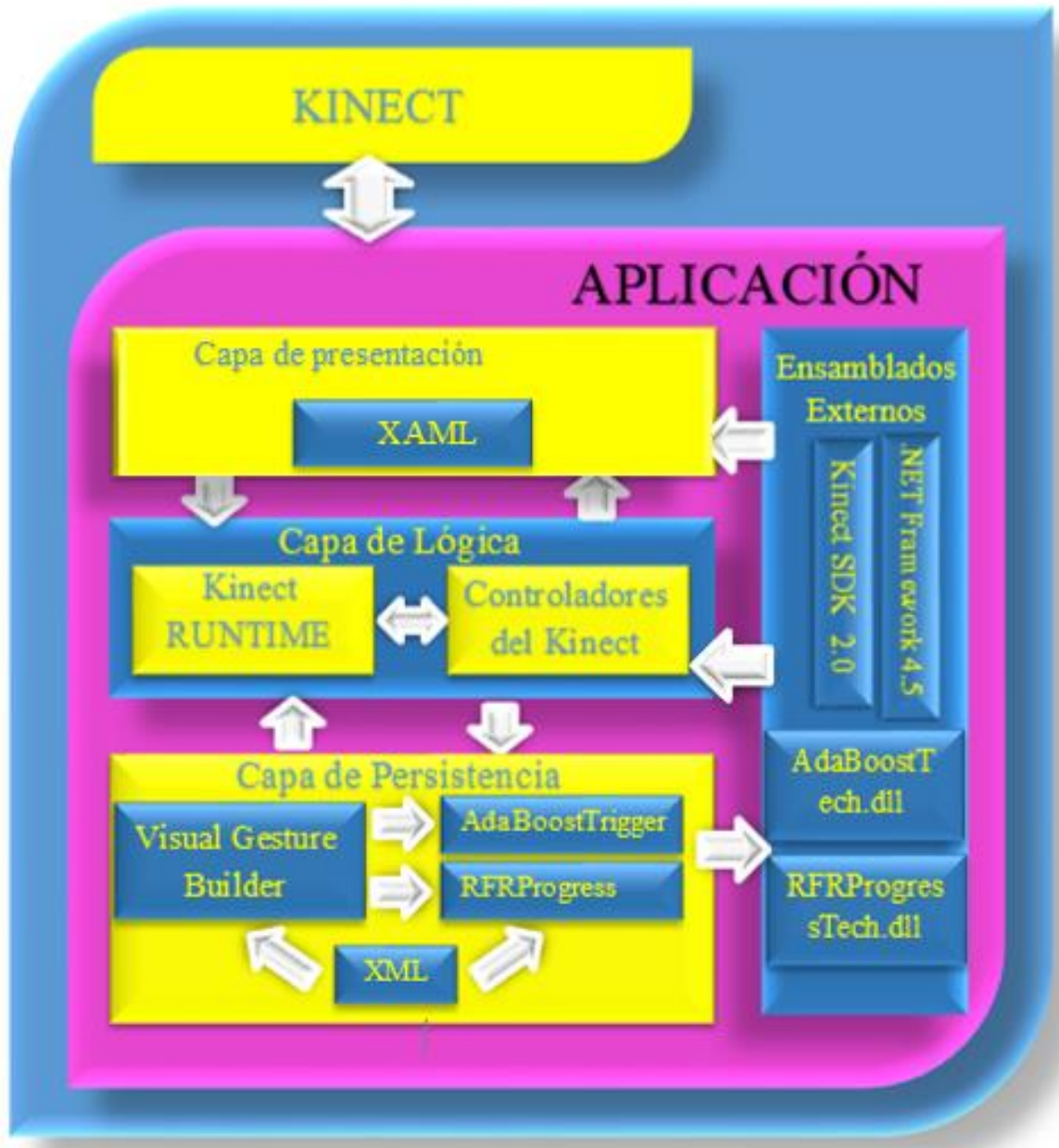


Figura 11. Arquitectura del sistema.⁴⁴

La aplicación está desarrollada bajo una configuración de 5 capas o niveles.

⁴⁴ Figura tomada de: Creación propia

En Primera instancia Tenemos el dispositivo que es considerado como la capa física, siendo la capa de más bajo nivel, está compuesta por el Kinect, pues es el hardware contiene los sensores y cámaras.

Capa lógica: La cual se divide en los controladores que permiten digitalizar los datos de los sensores que posibilitan su programación y el runtime que funciona como conexión entre la información que proporciona el hardware de Kinect y las diferentes APIs.

capa de persistencia: la cual tiene como función la administración y el acceso de los datos, posee un gestor de bases de datos en el cual se encuentran almacenados los parámetros y la información correspondiente a cada tipo de gestos soportados por dicho gestor, esta soportada por el Gesto Constructor Visual (VGB) y el lenguaje XML. Responde a solicitudes de lectura de datos desde la capa lógica.

Dentro del desarrollo del sistema es donde se demuestra por que al momento de escoger la tecnología que más beneficios presentara a la realización del sistema se escogió la versión dos de Kinect.

Durante el transcurso de esta etapa se utilizó la herramienta Kinect Studio v2.0 para realizar el proceso de aprendizaje, por medio del cual se le enseñó al sistema con qué tipo de gestos iba a trabajar, este proceso consistió en la elaboración de una serie de clips de video especializados para cada uno de los gestos que se quería incluir en el sistema, para establecer un nivel de confianza bastante fue necesario elaborar 20 clips de video por cada uno de los gestos.

En la figura 12 se muestra la realización del proceso de grabación de uno de estos clips de video.

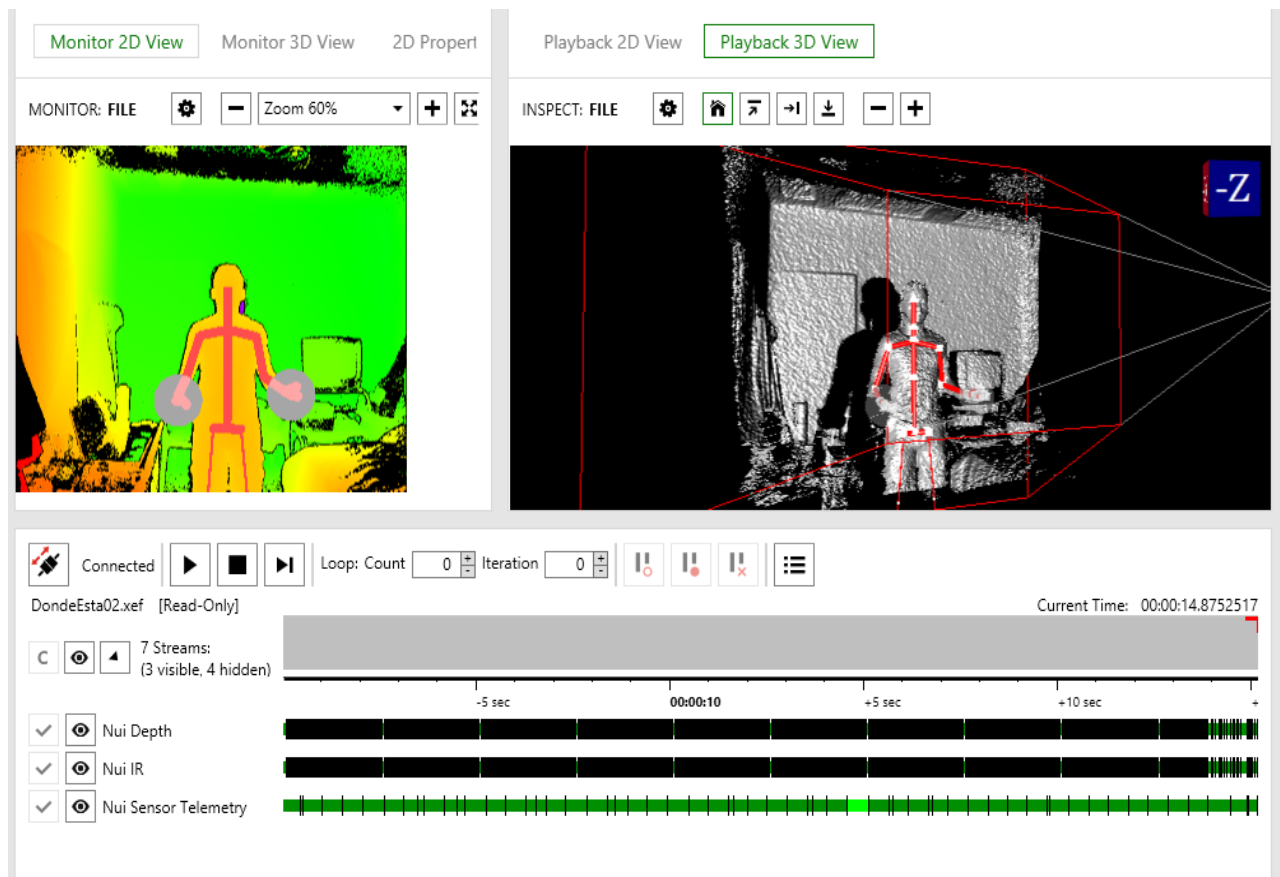
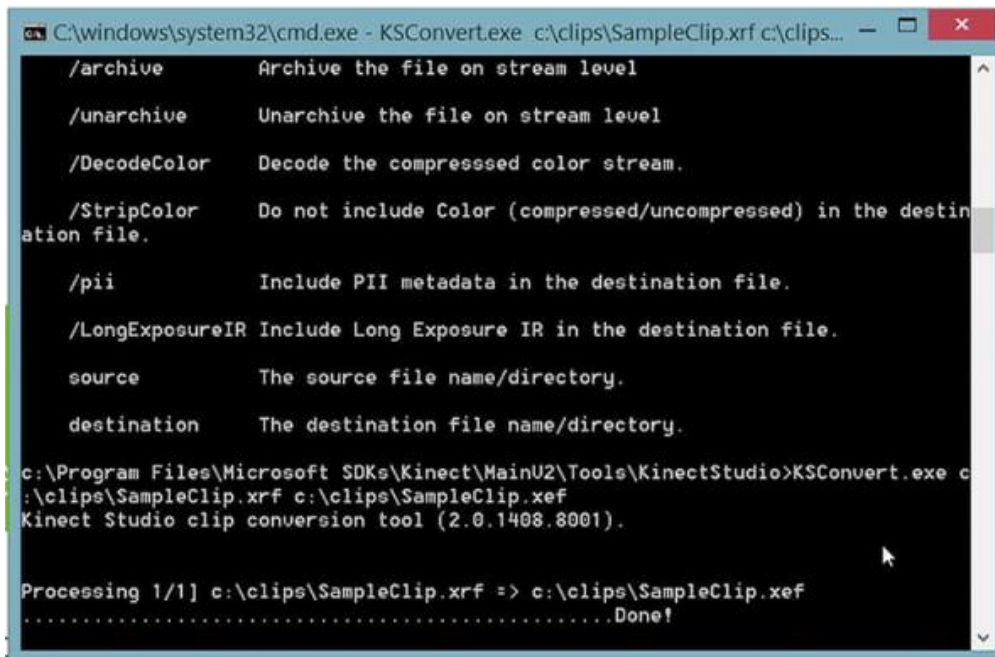


Figura 12. Grabando clips de video⁴⁵.

Luego de grabar cada uno de los clips pertinentes para cada gesto fue necesario aplicarles una conversión ya que de la forma en que los metadatos de los clips de video quedaban guardados no eran compatibles con Visual gesture builder y por ende no se iban a poder crear los archivos de bases de datos.

⁴⁵ Figura tomada de: creación propia.



```
C:\windows\system32\cmd.exe - KSConvert.exe c:\clips\SampleClip.xrf c:\clips\SampleClip.xef

/archive      Archive the file on stream level
/unarchive    Unarchive the file on stream level
/DecodeColor  Decode the compressed color stream.
/StripColor   Do not include Color (compressed/uncompressed) in the destination file.
/pii          Include PII metadata in the destination file.
/LongExposureIR Include Long Exposure IR in the destination file.

source        The source file name/directory.
destination   The destination file name/directory.

c:\Program Files\Microsoft SDKs\Kinect\MainU2\Tools\KinectStudio>KSConvert.exe c:\clips\SampleClip.xrf c:\clips\SampleClip.xef
Kinect Studio clip conversion tool (2.0.1408.8001).

Processing 1/1] c:\clips\SampleClip.xrf => c:\clips\SampleClip.xef
.....Done!
```

Figura 13. Conversión de clips de video⁴⁶.

Seguido a esto se utilizó la herramienta Visual Gesture Builder (VGB) para crear las bases de datos de los gestos, lo que llevo a un nuevo proceso de desarrollo el cual consistió en extraer de los clips de video previamente convertidos a un formato compatible una serie de datos gestuales de tipo discreto los cuales servían para enseñarle al sistema que segmentos específicos de cada clip de video se iba a considerar como un gesto valido.

En la figura 14 se muestra como se realizó esa extracción de información desde los clips de video, especificándole al sistema que tipo de gesto es y los segmentos del clip que si serán validados como gesto.

⁴⁶ Figura tomada de: creación propia.

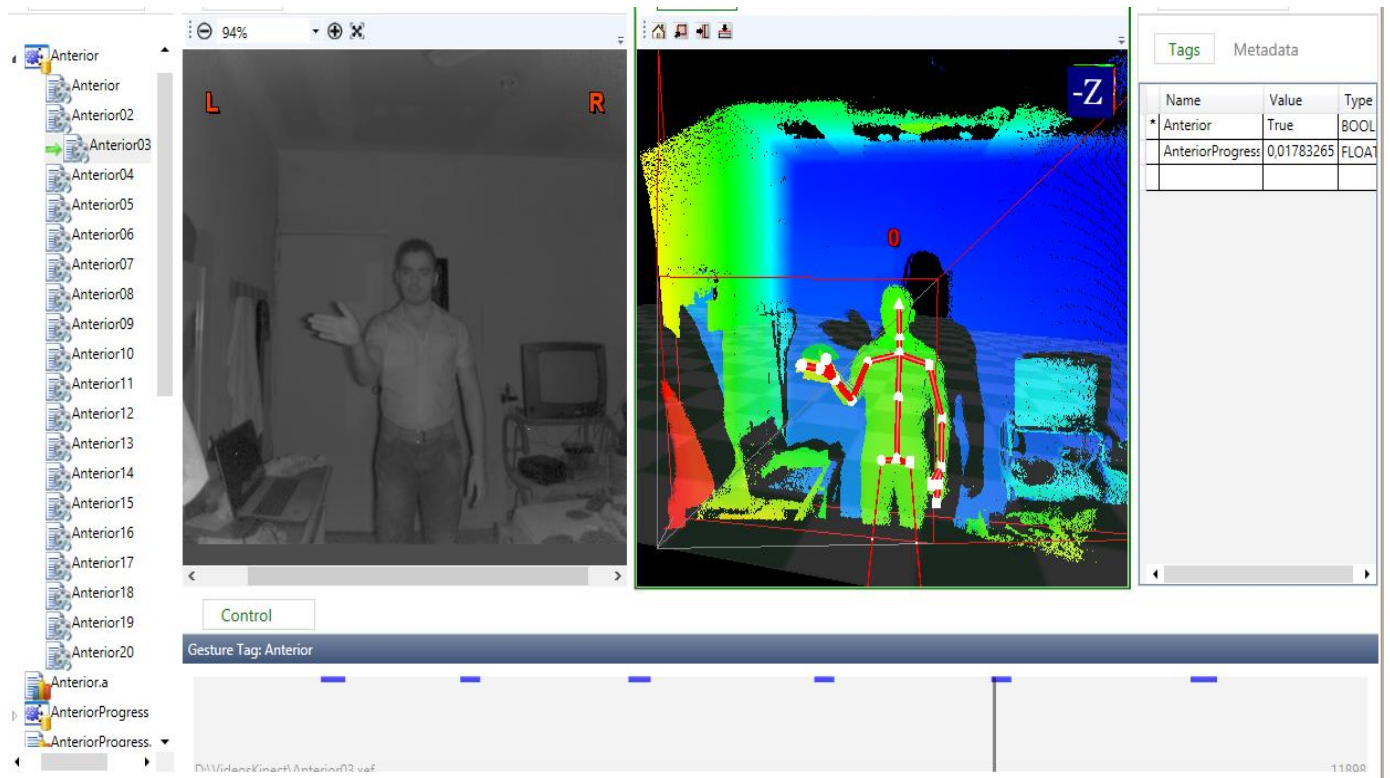


Figura 14. Enseñándole al sistema que gestos serán válidos

Terminados estos procesos se procedió a asociar los archivos de bases de datos a la aplicación para de esta manera asociarlos con los algoritmos de reconocimiento e interpretación realizados bajo el lenguaje de programación C#.

Una esquematización grafica de estos procesos descritos anteriormente se muestran en la siguiente figura.

Por ultimo pero no menos importante en el nivel más alto encontramos la capa de presentación, la cual comprende la interfaz del usuario, con la cual interactúa, esta le proporciona y solicita información. Posee características como ser entendible y fácil de usar, esta soportada por el lenguaje XAM. Esta capa solo se comunica con la capa de enlace quien a su vez se encarga de comunicarla con la capa lógica.

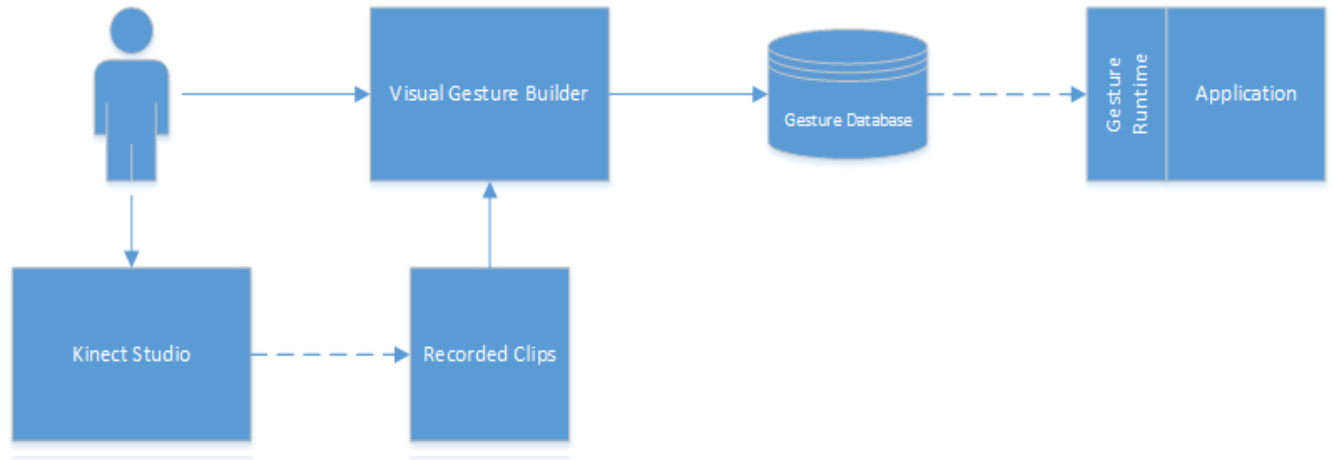


Figura 15. Diagrama de la estructura del desarrollo⁴⁷.

5.2. Diseño del sistema

Se procede a realizar una definición simple y concisa los procesos funcionales al igual que las restricciones pertinentes al sistema Traductor de sordo, a raíz de este diseño del sistema se inicia la definición de lo casos de uso funcionales y no funcionales, establecer cuáles son los objetos dentro del sistema, elaborar una descripción detallada de los actores, y la realización de los diagramas correspondientes a la interacción entre los actores y el sistema.

El levantamiento de los requisitos para el desarrollo de este sistema fundamenta sus funciones y basa su elaboración en los criterios dados en el estándar IEEE 830 y en el modelo de caso de uso. Hacer uso de esta jerarquía de datos, es decir, seguir estos estándares permite que la definición y descripción de los requisitos tanto funcionales como no funcionales sea mucho más fácil y más coherente.

⁴⁷ Figura tomada de: msdn.microsoft.com/en-us/library/dn785529.aspx, fecha de consulta: 2014

5.2.1. Especificación de requerimientos

Para cualquier proyecto, la especificación de requisitos debe ser considerado como una de las actividades más significativas y más importantes, ya que apoyándose en estos requerimientos es que procedemos a tomar medidas acerca de cómo se va a hacer el desarrollo, este también simplifica la determinación de necesidades, la definición de características y el alcance de la solución, además de la construcción de modelos (comportamiento y secuencial) del sistema, este proceso ayuda a minimizar los problemas relacionados con el desarrollo.

El objetivo de la especificación de requerimientos es actuar como agente controlador estableciendo funcionalidades, realizando análisis y definiendo las excepciones a las cuales va a responder el sistema. A parte de esto para realizar la especificación de los requerimientos se hace necesaria la elaboración de un documento que exponga de forma detallada el comportamiento así como la definición del conjunto de requerimientos del sistema a desarrollar.

5.2.2. Objetivos del sistema

Ahora se procederá a detallar los objetivos del sistema Traductor de Sordomudos, cada uno será descrito en las tablas 3 y 4 respectivamente.

OBJ – 1	Realización del Gesto.
Descripción	<p>El Traductor de Sordomudos permitirá calibrar e identificar el cuerpo de los usuarios por medio de un algoritmo de detección de gestos.</p> <p>Aquel usuario que no esté a una distancia adecuada del sensor no podrá ser reconocido como tal y por ende su cuerpo no será identificado.</p>
Estabilidad	Alta
Comentarios	El sistema podrá reconocer a cualquier usuario sin necesidad tener que agregarlo, ya que al detectar un nuevo cuerpo el sistema se calibrará a las características de este nuevo cuerpo identificándolo como un usuario activo.

Tabla 2. Realización del gesto⁴⁸

OBJ – 2	Traducción de los Gestos.
Descripción	<p>El sistema Traductor de Sordomudos tendrá un área de reconocimiento, a través de la cual recibirá una serie de información gestual proporcionada por el usuario por medio de una interfaz de usuario natural (NUI). Devolviendo traducciones textuales al usuario, en respuesta al gesto realizado por el mismo.</p> <p>El sistema podrá realizar traducción de gestos a 6 usuarios a mismo tiempo.</p>
Estabilidad	Alta
Comentarios	El sistema puede detectar un máximo de 6 usuarios al tiempo, es decir todos pueden estar en el área de

⁴⁸ Tabla tomada de: Creación propia

	reconocimiento al tiempo, pero la traducción de gestos se realiza un usuario a la vez.
--	--

Tabla 3. Traducción de los Gestos⁴⁹.

5.2.3. Diagrama de casos de uso

Los diagramas de casos de uso no son más que una representación gráfica del funcionamiento interno de un sistema y de la forma en como dicho sistema se relaciona con los diferentes agentes que componen directa e indirectamente su funcionamiento. En el caso del sistema Traductor de Sordomudos, su funcionamiento se resume en una estructura de diagramas específica, la cual veremos a continuación en la figura 26.

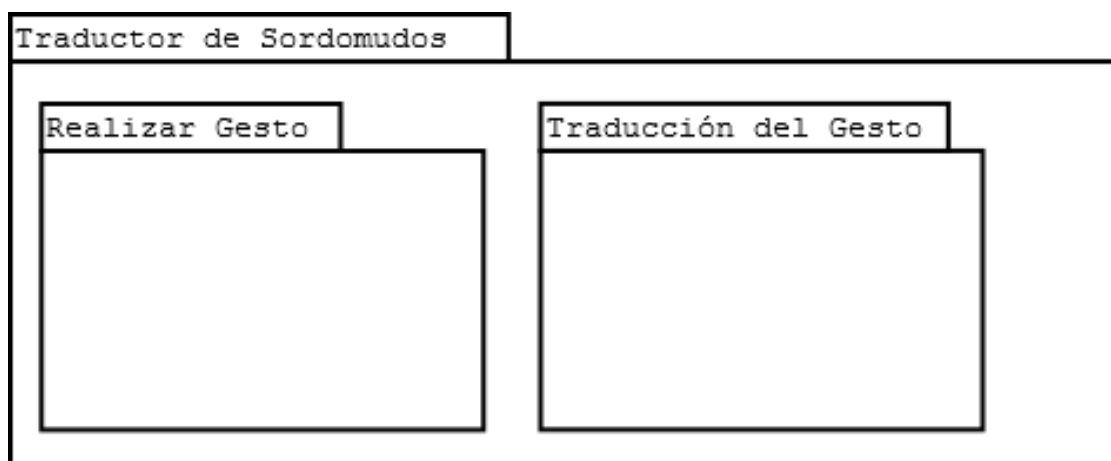


Figura 26. Diagrama de Subsistemas⁵⁰.

⁴⁹ Tabla tomada de: Creación propia.

⁵⁰ Imagen tomada de: creación propia

5.2.3.1. Diagrama de caso de uso realizar gesto

A continuación se muestra gráficamente por medio de la figura 27, el escenario realizar gesto, en este se detalla el funcionamiento interno del sistema en el momento en el cual el usuario realiza los gestos que posteriormente serán interpretados, además de eso también se indica la necesidad primordial de reconocer el cuerpo del usuario, para que este pueda cumplir con la acción en cuestión.

Dentro del desarrollo de este escenario, contamos con la participación de dos actores, un usuario y un sensor Kinect, también contamos con tres casos de uso los cuales a su vez poseen sus respectivas interrelaciones.

Ahora procedemos ver el diagrama de caso del caso de uso realizar gesto.

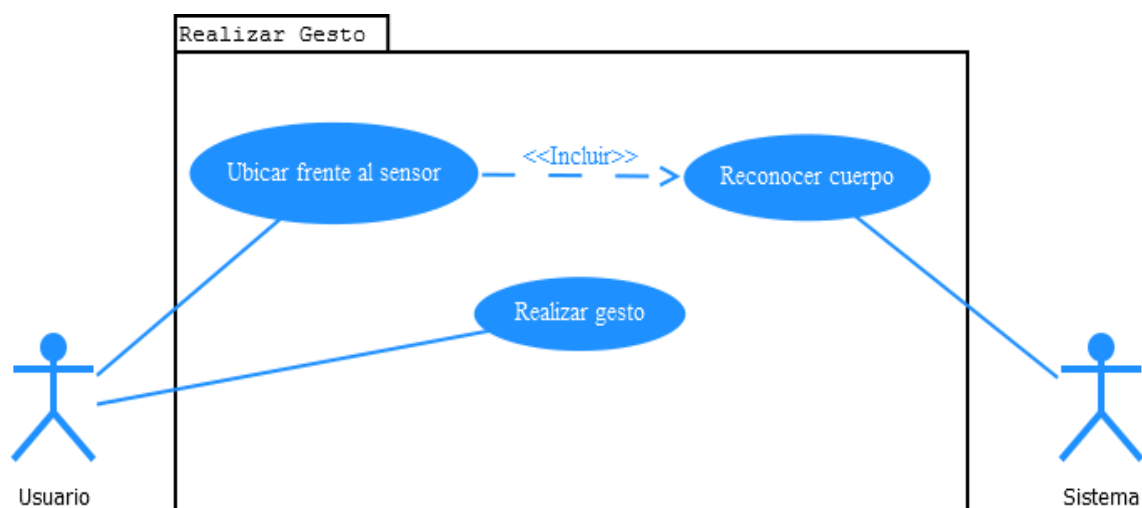


Figura 27. Caso de uso realizar gesto⁵¹.

⁵¹ Imagen tomada de: creación propia

5.2.3.2. Diagrama de caso de uso traducción del gesto

A continuación se muestra gráficamente por medio de la figura 28, el escenario traducción del gesto, en este se detalla el funcionamiento interno del sistema en el momento en el cual el sistema interpreta los gestos que el usuario realiza.

Dentro del desarrollo de este escenario, contamos con la participación de tres actores, un usuario, un sensor Kinect y una base de datos, también contamos con cinco casos de uso los cuales a su vez poseen sus respectivas interrelaciones.

Ahora procedemos ver el diagrama de caso del caso de uso traducción del gesto.

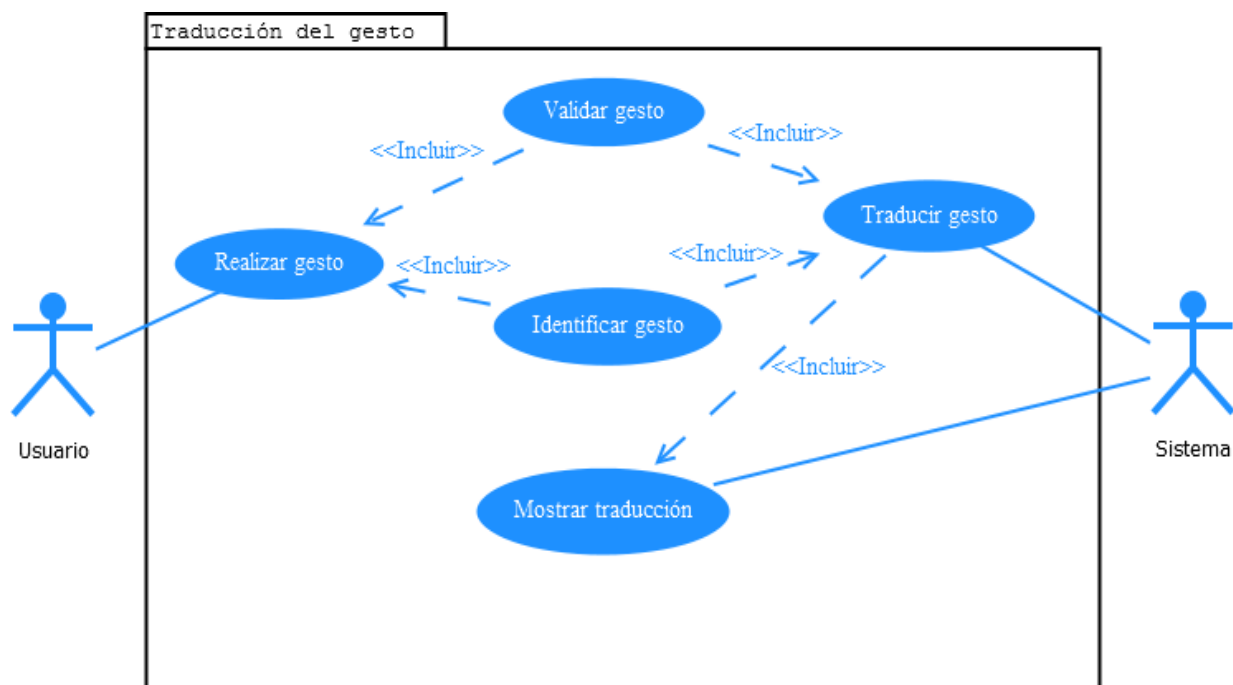


Figura 28. Caso de uso Traducción del gesto⁵².

⁵² Imagen tomada de: creación propia.

5.2.4. Diagramas de secuencia

Los diagramas de secuencia son un tipo de diagramas que usamos para mostrar las relaciones existentes entre los diferentes objetos que conforman un sistema, cada una de las relaciones con su respectiva duración a través del tiempo, también puede verse como una secuencia de mensajes los cuales pueden ser peticiones o respuestas enviados entre los diferentes objetos que componen un sistema.

5.2.4.1. Diagrama de secuencia de realizar gesto

En este diagrama de secuencia se definen paso a paso las acciones que se deben realizar tanto a nivel usuario, como a nivel sistema para permitir que un usuario pueda realizar un gesto, de igual forma también se muestra como el sistema internamente reconoce un cuerpo como un nuevo usuario.

A continuación veremos el grafico que representa este diagrama de secuencia.

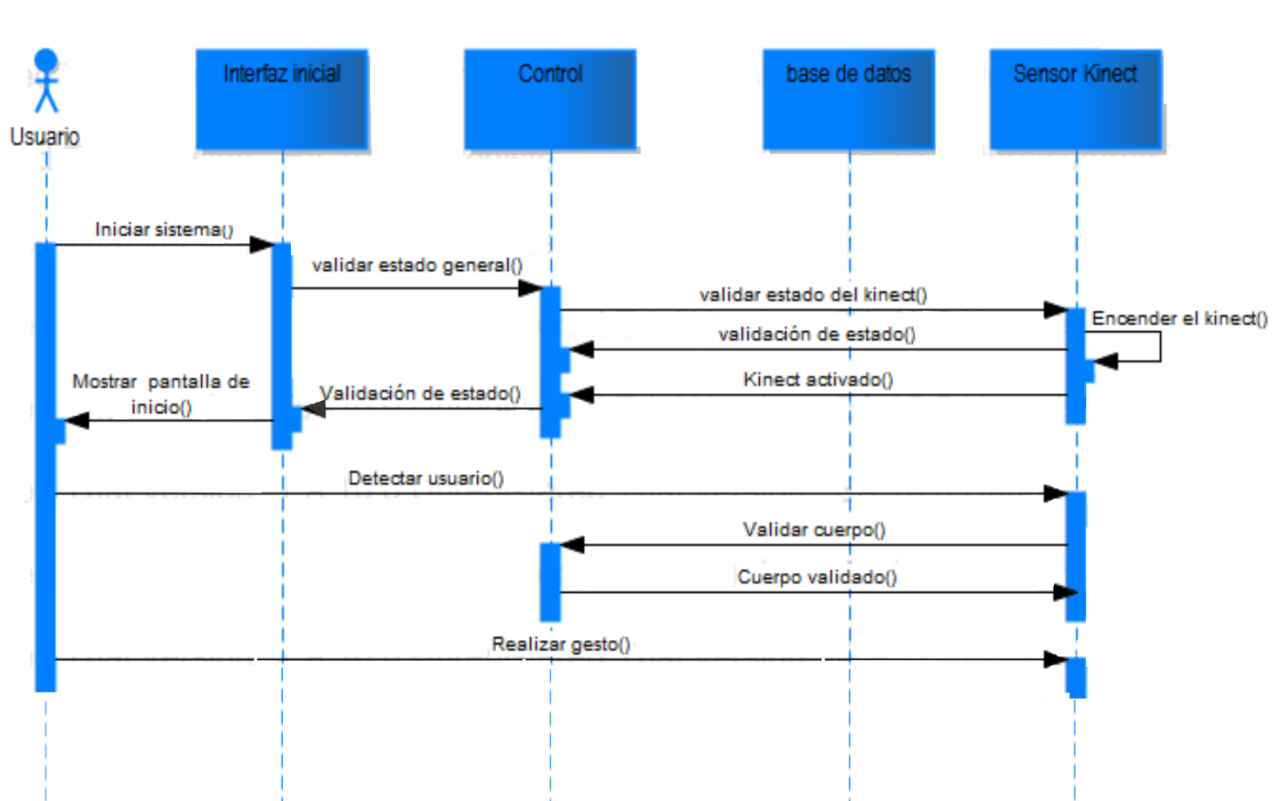


Figura 29. Diagrama de secuencia realizar gesto⁵³.

5.2.4.2. Diagrama de secuencia traducción del gesto

En este diagrama de secuencia se definen paso a paso las acciones que se deben realizar tanto a nivel usuario, como a nivel sistema para permitir que el sistema valide e identifique los gestos realizados por un usuario para posteriormente proceder a traducir dicho gesto en un mensaje que el usuario podrá ver en pantalla.

A continuación veremos el grafico que representa este diagrama de secuencia.

⁵³ Imagen tomada de: creación propia.

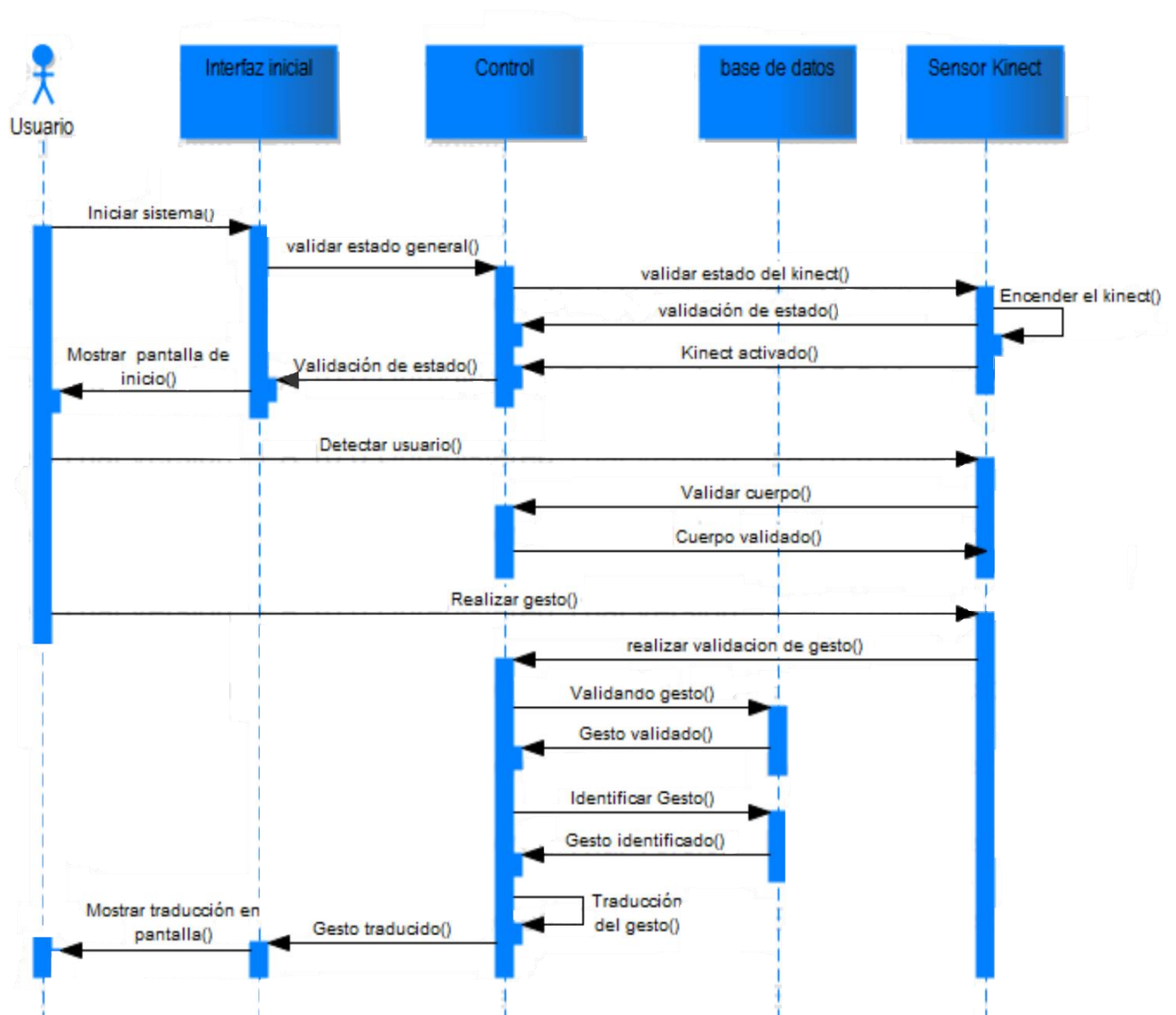


Figura 30. Diagrama de secuencia traducción del gesto⁵⁴.

5.2.5. Diagrama de actividades

Este diagrama nos permite divisar el flujo de las actividades realizadas desde el punto inicial hasta el punto final, mostrándonos la red de decisiones que se originan en el progreso de los eventos contenidos dentro del desarrollo de un proceso. Estos diagramas

⁵⁴ Imagen tomada de: creación propia.

son muy útiles a la hora de proyectar a resultado te llevara tomar una decisión o no tomarla.

5.2.5.1. Diagrama de actividades realizar gesto

El diagrama de actividades para la acción realizar gesto especifica varias actividades o en este caso decisiones que un usuario debe tomar para poder realizar un gesto.

En este diagrama de clases se muestra un flujo de actividades acerca de 3 decisiones que transcurren entre el momento en que se inicia el sistema, hasta que se cumple el objetivo de este. En la primera decisión el usuario se ubica frente al sensor pero su cuerpo no fue calibrado correctamente; En la segunda, el usuario se ubica frente al sensor y su cuerpo si es calibrado y por último en donde el sistema está en espera de gestos pero el usuario no se ubica frente al sensor.

En la siguiente imagen se evidencia el diagrama de actividad para realizar gesto.

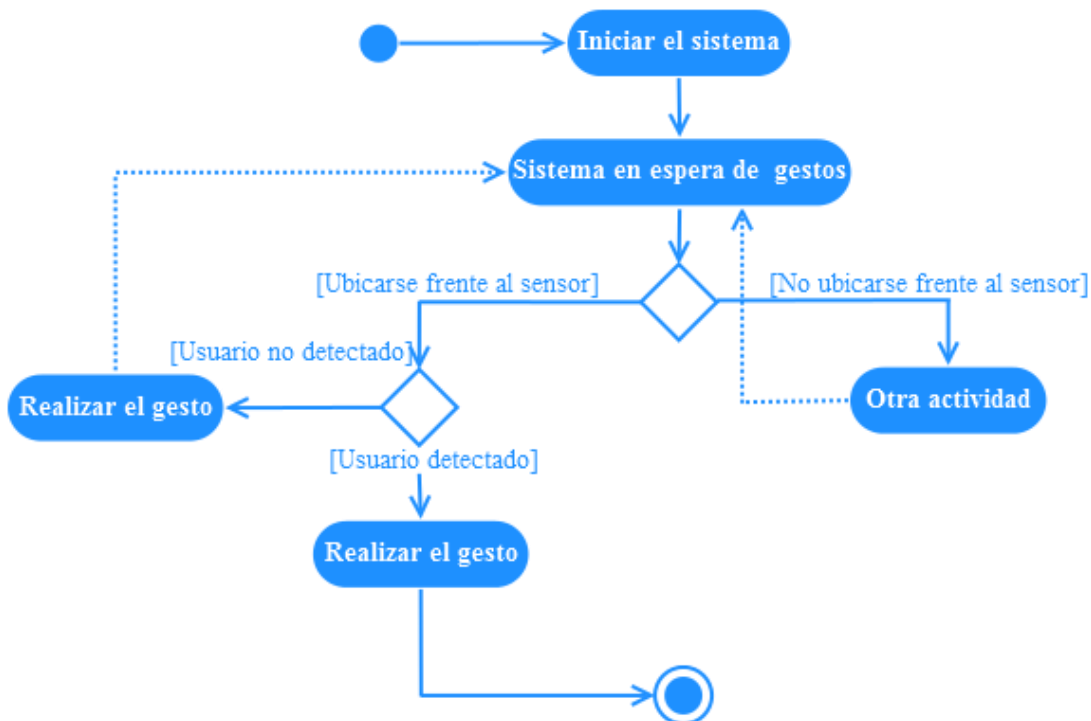


Figura 31. Diagrama de actividades realizar gesto⁵⁵.

⁵⁵ Imagen tomada de: creación propia.

5.2.5.2. Diagrama de actividades traducción del gesto

El diagrama de actividades para la acción traducción del gesto especifica varias actividades o en este caso decisiones que un usuario debe tomar para que el sistema sea capaz de traducir sus gestos.

En este diagrama de clases se muestra un flujo de actividades acerca de 6 decisiones que transcurren entre el momento en que se inicia el sistema, hasta que se cumple el objetivo de este. Al igual que en diagrama anterior (Figura 31), en la primera decisión el usuario se ubica frente al sensor pero su cuerpo no fue calibrado correctamente; En la segunda, el usuario se ubica frente al sensor y su cuerpo si es calibrado; En tercer lugar también, en donde el sistema está en espera de gestos pero el usuario no se ubica frente al sensor. A demás de las decisiones anteriores que ya se evidenciaron en el caso de uso anterior en este también se muestra cuando el usuario realiza un gesto con su cuerpo ya calibrado pero este gesto no es válido y también cuando el usuario con su cuerpo ya calibrado realiza un gesto que si es válido el cual procede a ser traducido.

En la siguiente imagen se evidencia el diagrama de actividad para traducción del gesto.

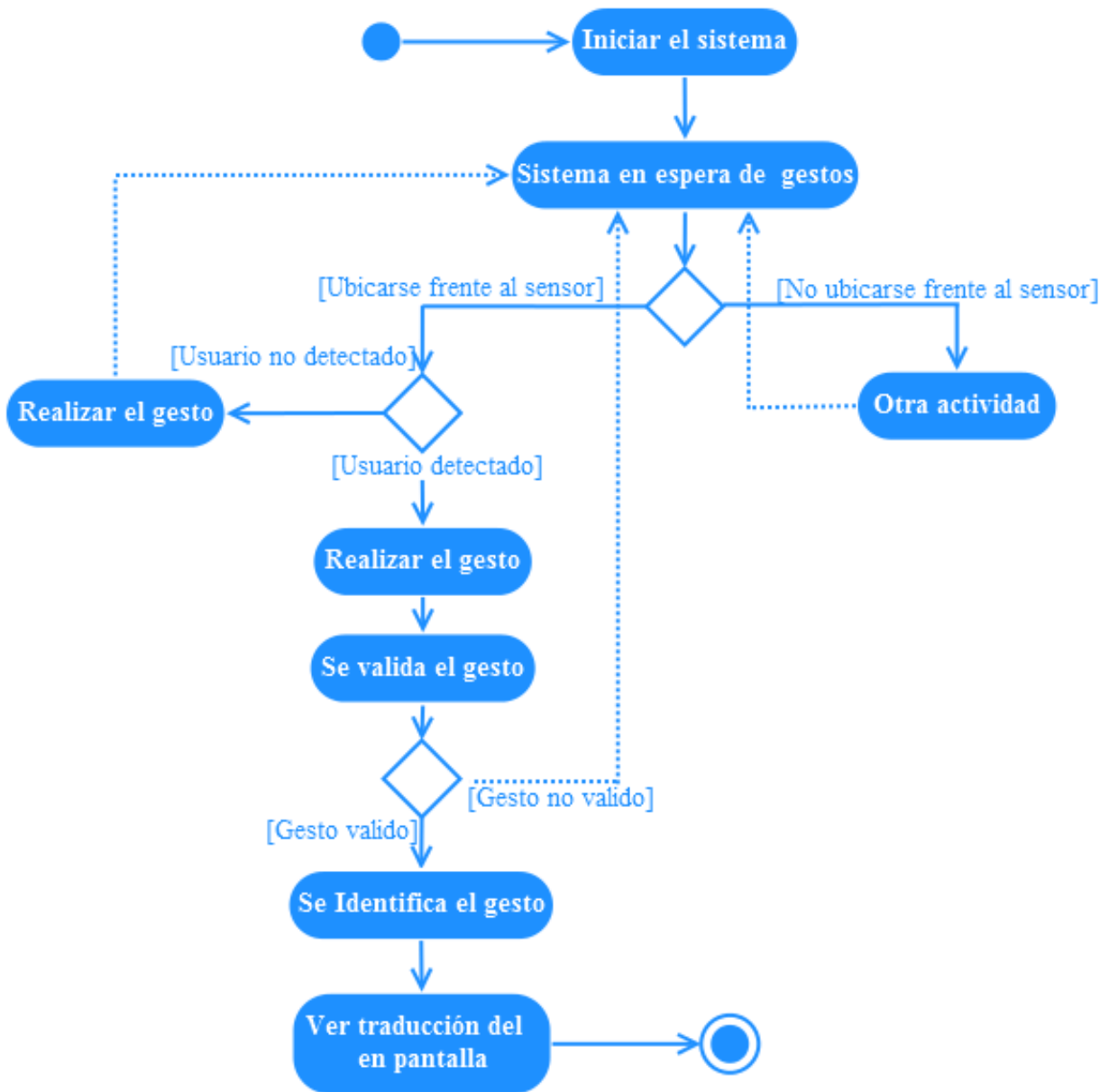


Figura 32. Diagrama de actividades traducción del gesto⁵⁶.

⁵⁶ Imagen tomada de: creación propia.

5.2.6. Requisitos de información

En este apartado se procederá a definir los requisitos de información pertinentes para el proyecto Traductor de Sordomudos.

RI - 01	Información sobre la realización de gestos
Objetivos asociados	OBJ-01
Requisitos asociados	RF-01: Ubicarse frente al sensor. RF-02: Reconocer cuerpo. RF-03: Realizar el gesto
Descripción	El sistema debe calibrar el cuerpo del usuario que lo va a usar antes de iniciar la recepción de los gestos.
Datos específicos	Estado del cuerpo
Tiempo de vida promedio	
Ocurrencias	Alta
Comentarios	Ninguno

Tabla 4. Requisitos de información sobre realización de gesto⁵⁷.

RI - 02	Información sobre la traducción de gestos
Objetivos asociados	OBJ-02
Requisitos asociados	RF-01: Ubicarse frente al sensor RF-02: Reconocer cuerpo como un nuevo usuario. RF-03: Realizar gesto RF-04: Validar gesto RF-05: Identificar gesto RF-06: Traducir gesto

⁵⁷ Tabla tomada de: creación propia

Descripción	El sistema deberá ser capaz de traducir los gestos realizados por el usuario cuyo cuerpo fue previamente calibrado.
Datos específicos	Estado del cuerpo datos del gesto
Tiempo de vida promedio	
Ocurrencias	Alta
Comentarios	Ninguno

Tabla 5. Requisitos de información sobre traducción de gestos⁵⁸.

5.2.7. Documentación de los actores

A continuación se procede a describir en cada una de las tablas siguientes a cada uno de los actores que interactúa directa o indirectamente con el sistema.

Actor	Usuario
Casos de uso	Ubicarse frente al sensor, realizar el gesto.
Descripción	Este actor está representado por los usuarios que utilizarán el sistema y que tendrán la oportunidad de comunicarse a través del lenguaje de señas colombiano. El cuerpo del actor debe ser calibrado antes de que el gesto que realice pueda ser traducido.

Tabla 6. Información sobre el actor: Usuario⁵⁹.

⁵⁸ Tabla tomada de: creación propia

⁵⁹ Tabla tomada de: creación propia.

Actor	Sistema
Casos de uso	Reconocer cuerpo, validar gesto, identificar gesto, traducir el gesto, mostrar traducción.
Descripción	Este actor es el que controla y administra los procesos, es el encargado de reconocer el cuerpo de una persona como un nuevo usuario, valida los gestos a ver si son permitidos, luego de validarlos busca en la base de datos la identificación textual de dicho gesto para posteriormente realizar la traducción del mismo y por ultimo mostrar dicha traducción en pantalla.

Tabla 7. Información sobre el actor: Sistema⁶⁰.

5.2.8. Documentación de los casos de uso

En esta sección se realiza una especificación muy detallada de todos y cada uno de los casos de uso expuestos gráficamente en secciones anteriores (ver diagramas de casos de uso), los cuales muestran las interrelaciones entre el sistema y el usuario y viceversa.

⁶⁰ Tabla tomada de: creación propia.

RF - 01	Ubicarse frente al sensor	
Objetivos asociados	OBJ – 1 Realización del gesto.	
Requisitos asociados	RI – 01 Información sobre la realización del gesto. RI – 02 Información sobre la traducción del gesto.	
Descripción	En este caso de uso se muestra como se manejan los procesos internos del sistema cuando el usuario debe ubicarse dentro del rango de detección del sensor Kinect.	
Precondición	Para que el usuario pueda proceder a ubicarse frente al sensor, el sistema ya debe encontrarse a la espera de la detección.	
Secuencia normal	Paso	Acción
	1	Se inicia el sistema.
	2	El sistema está a la espera de la detección.
	3	El usuario se ubica frente al Kinect.
Poscondición	Después de que el usuario se ubica en frente del sensor, el sistema iniciara los procesos necesarios para reconocer el cuerpo del usuario.	
Excepciones	El usuario se ubica frente al sensor pero el sistema no se encuentra a la espera de la detección de usuarios (él no ha sido iniciado).	
	Paso	Acción
	1	El usuario no inicia el sistema.
	2	El usuario se ubica frente al sensor
Rendimiento		
Frecuencia esperada		
Importancia	Alta	
Comentarios	Si el usuario no se ubica dentro del rango de reconocimiento del sensor Kinect, será imposible reconocerlo como usuario.	

Tabla 8. Documentación del caso de uso ubicarse frente al sensor⁶¹.

⁶¹ Tabla tomada de: creación propia.

RF - 02	Reconocer cuerpo	
Objetivos asociados	OBJ – 1 Realización del gesto.	
Requisitos asociados	RI – 01 Información sobre la realización del gesto. RI – 02 Información sobre la traducción del gesto.	
Descripción	En este caso de uso se muestra como se manejan los procesos internos del sistema a la hora de reconocer el cuerpo de un nuevo usuario.	
Precondición	Para que el cuerpo del usuario pueda ser reconocido, el sistema ya debe estar a la espera del reconocimiento de un nuevo cuerpo.	
Secuencia normal	Paso	Acción
	1	El sistema está a la espera del reconocimiento del cuerpo.
	2	El usuario se ubica frente al Kinect.
	3	El sistema calibra su cuerpo y lo reconoce como usuario.
Poscondición	Después de realizar el reconocimiento del nuevo cuerpo, el usuario podrá realizar la siguiente actividad.	
Excepciones	Escenario 1: No ubicarse frente al sensor	
	Paso	Acción
	1	El usuario no se ubica frente al sensor.
	2	El sistema queda en espera del reconocimiento del cuerpo.
	Escenario 2: El cuerpo del usuario no fue detectado.	
	Paso	Acción
	1	El usuario se ubica a una distancia inadecuada del sensor.
	2	El sistema no reconoce el cuerpo del nuevo usuario.

	3	El sistema queda en espera del reconocimiento del cuerpo del nuevo usuario.
Rendimiento		
Frecuencia esperada		
Importancia	Alta	
Comentarios	La detección del cuerpo es primordial para que el usuario puede ser definido como tal y pueda proceder a realizar los gestos que deseé.	

Tabla 9. Documentación del caso de uso reconocer cuerpo⁶²

RF - 03	Realizar gesto	
Objetivos asociados	OBJ – 1 Realización del gesto.	
Requisitos asociados	RI – 01 Información sobre la realización del gesto. RI – 02 Información sobre la traducción del gesto.	
Descripción	En este caso de uso se muestra como se manejan los procesos internos del sistema a la hora de que el usuario realiza un nuevo gesto.	
Precondición	Para que el usuario pueda proceder a realizar el gesto que desea, primero debe ubicarse frente al sensor para que este reconozca su cuerpo catalogándolo como un usuario valido.	
Secuencia normal	Paso	Acción
	1	El sistema está a la espera del reconocimiento de gestos.
	2	El usuario se ubica frente al Kinect.
	3	El sistema calibra su cuerpo y lo reconoce como un nuevo usuario.
	4	El usuario puede realizar el gesto que deseé.

⁶² Tabla tomada de: creación propia.

Poscondición	Después de realizar el gesto el sistema procederá a realizar una serie de procesos para continuar con la siguiente actividad.	
Excepciones	Escenario 1: El usuario realiza un gesto inválido	
	Paso	Acción
	1	El usuario se ubica frente al sensor.
	2	El sistema reconoce el nuevo cuerpo validándolo como un usuario.
	3	El usuario realiza un gesto inválido.
	4	El sistema queda a la espera de la realización de un gesto válido.
	Escenario 3: El usuario realiza un gesto valido pero su cuerpo no había sido calibrado.	
	Paso	Acción
	1	El usuario se ubica a una distancia inadecuada del sensor.
	2	El sistema no reconoce el cuerpo del nuevo usuario.
	3	El usuario realiza el gesto.
	4	El sistema queda en espera del reconocimiento del cuerpo del nuevo usuario.
Rendimiento		
Frecuencia esperada		
Importancia	Alta	
Comentarios	La realización de un gesto es información más importante para el sistema, ya que el gesto al ser detectado, desencadena la función principal del sistema.	

Tabla 10. Documentación del caso de uso reconocer el cuerpo⁶³.

⁶³ Tabla tomada de: creación propia.

RF - 04	Validar gesto	
Objetivos asociados	OBJ – 2 Traducción del gesto.	
Requisitos asociados	RI – 01 Información sobre la realización del gesto. RI – 02 Información sobre la traducción del gesto.	
Descripción	En este caso de uso se muestra como se manejan los procesos internos del sistema a la hora de validar un gesto realizado por el usuario.	
Precondición	Para que el sistema pueda proceder a validar un gesto que realizado por el usuario, primero el usuario debe realizar un gesto.	
Secuencia normal	Paso	Acción
	1	El sistema procede a validar el gesto realizado.
	2	El sistema consulta los gestos asociados a la base de datos.
	3	El sistema valida el gesto.
Poscondición	Después de validar el gesto el sistema procederá a realizar una serie de procesos para continuar con la identificación del mismo.	
Excepciones	El usuario realiza un gesto inválido	
	Paso	Acción
	1	El usuario realiza un gesto inválido.
	2	El sistema verifica el gesto dentro de los gestos asociados a la base de datos.
	3	El sistema valida que es un gesto no válido
	4	El sistema queda a la espera de la realización de un gesto válido.
Rendimiento		
Frecuencia esperada		

Importancia	Alta
Comentarios	La validación del gesto, se realiza para no tener que ingresar a la base de datos por cada posible gesto que sea detectado por el sensor, si no para que ingrese a la base de datos cuando sea seguro que va a existir un gesto asociado.

Tabla 11. Documentación del caso de uso validar gesto⁶⁴.

RF - 05	Identificar gesto	
Objetivos asociados	OBJ – 2 Traducción del gesto.	
Requisitos asociados	RI – 01 Información sobre la realización del gesto. RI – 02 Información sobre la traducción del gesto.	
Descripción	En este caso de uso se muestra como se manejan los procesos internos del sistema a la hora de identificar un gesto realizado por el usuario.	
Precondición	Para que el sistema pueda proceder a identificar un gesto que realizado por el usuario, primero dicho gesto debe haber superado el proceso de validación.	
Secuencia normal	Paso	Acción
	1	El sistema procede a identificar el gesto ya validado.
	2	El sistema consulta directamente dentro de la base de datos y selecciona los datos correspondientes al gesto previamente validado.
	3	El sistema identifica el gesto.
Poscondición	Después de identificar el gesto, el sistema procederá a realizar una serie de procesos para continuar con la traducción del mismo.	

⁶⁴ Tabla tomada de: creación propia.

Excepciones	
Rendimiento	
Frecuencia esperada	
Importancia	Alta
Comentarios	La identificación del gesto, se realiza solamente después de haber validado la existencia del mismo dentro de la base de datos, con el fin de maximizar el tiempo de respuesta y optimizar recursos.

Tabla 12. Documentación del caso de uso identificar gesto⁶⁵.

RF - 06	Traducir gesto	
Objetivos asociados	OBJ – 2 Traducción del gesto.	
Requisitos asociados	RI – 02 Información sobre la traducción del gesto.	
Descripción	En este caso de uso se muestra como se manejan los procesos internos del sistema a la hora de traducir un gesto realizado por el usuario.	
Precondición	Para que el sistema pueda proceder a traducir un gesto que realizado por el usuario, primero dicho gesto debe haber superado el proceso de identificación, para que los datos del gesto provenientes de la base de datos ya se encuentren cargados en el sistema en tiempo de ejecución.	
Secuencia normal	Paso	Acción
	1	El sistema procede a traducir el gesto que fue identificado anteriormente.

⁶⁵ Tabla tomada de: creación propia.

	2	El sistema carga toda la información del gesto tomada desde la base de datos dentro de un algoritmo de traducción en tiempo de ejecución.
	3	El sistema identifica el gesto.
Poscondición	Después de tener una versión manipulable de los datos del gesto extraídos de la base de datos, el sistema procederá con la traducción del gesto en cuestión.	
Excepciones		
Rendimiento		
Frecuencia esperada		
Importancia	Alta	
Comentarios	La traducción del gesto, se realiza luego de que ya el sistema tiene almacenados todos los datos del gesto, para poder manipular todas las variables del mismo.	

Tabla 13. Documentación del caso de uso traducir gesto⁶⁶.

RF - 05	Mostrar traducción
Objetivos asociados	OBJ – 2 Traducción del gesto.
Requisitos asociados	RI – 02 Información sobre la traducción del gesto.
Descripción	En este caso de uso se evidencia como se manejan los procesos internos del sistema a la hora de mostrarle al usuario la traducción del gesto que ha pasado por todo el proceso del sistema.

⁶⁶ Tabla tomada de: creación propia.

Precondición	Para que el sistema pueda proceder a mostrar la traducción del gesto en pantalla, primero dicho gesto debe haber sido traducido.	
Secuencia normal	Paso	Acción
	1	El sistema procede a mostrar el gesto ya traducido en pantalla.
	2	El sistema manipula las variables obtenidas luego de traducir el gesto y toma la que tenga el formato correcto para mostrarla en un elemento de la pantalla.
	3	El sistema muestra la traducción del gesto.
Poscondición	Después de mostrar la traducción del gesto, el usuario puede entender lo que representa textualmente ese gesto en específico; y el sistema queda automáticamente en espera de recibir un nuevo gesto.	
Excepciones		
Rendimiento		
Frecuencia esperada		
Importancia	Alta	
Comentarios	El proceso de mostrar el gesto en pantalla, es simplemente tomar una variable del gesto cargada en el sistema desde la base de datos y asignársela a un elemento dentro de la interfaz.	

Tabla 14. Documentación del caso de uso mostrar traducción⁶⁷.

⁶⁷ Tabla tomada de: creación propia.

5.2.9. Requisitos no funcionales

5.2.9.1. Facilidad de uso

El uso de este sistema está diseñado bajo los criterios de una Interfaz Natural de Usuario (NUI), por ende no cuenta con un manual de usuario ya que el usuario que lo valla a utilizar solamente debe hablar naturalmente en el lenguaje de señas colombiano (vocabulario básico), el cual para el caso de los usuarios a los cuales está dirigido este sistema es una lengua que dominan a la perfección.

La única recomendación para el uso adecuado del sistema es que el usuario tenga presente que en esta versión del sistema solo se cuenta con la traducción del vocabulario básico.

5.2.9.2.Sistema no invasivo para el usuario

En el mundo entero no puede haber un tipo de software que presente mayor libertad para el usuario que un software desarrollado con tecnología de interfaz natural (NUI), ya que precisamente para eso está pensada esta tecnología, puesto que le permite al usuario interactuar con el sistema sin siquiera tocar una tecla, por lo que el usuario no estará sujeto a ninguna condición física para interactuar plenamente con el sistema.

5.2.9.3.Detección de errores

Este sistema está desarrollado con el propósito de que no alerte al usuario si este llega a realizar un gesto que no está permitido o que simplemente no exista dentro del contexto de la base de datos; Más bien está diseñado para reaccionar a la coincidencia entre el gesto realizado y el gesto almacenado en la base de datos, ya que al tratarse de una aplicación de Interfaz de usuario natural (NUI), la cual luego de reconocer a un nuevo usuario esta todo el tiempo a la espera de gestos, tan solo con realizar cualquier mínimo movimiento se le están enviando datos al sistema los cuales este podría tomar como

gestos realizados y entonces todo el tiempo el sistema nos estaría mandando mensajes de error antes de que pudiéramos realizar un gesto que si se encuentre dentro de la base de datos.

5.2.10. Diagrama de componentes

Un diagrama de componentes se encarga de esquematizar gráficamente la división del software mediante componentes al igual que las relaciones existentes entre ellos. El diagrama contiene, componentes, interfaces y relaciones, los elementos que contendrá pueden ser librerías, ejecutables, tablas, archivos y documentos, etc. Que estén incluidos en el sistema.

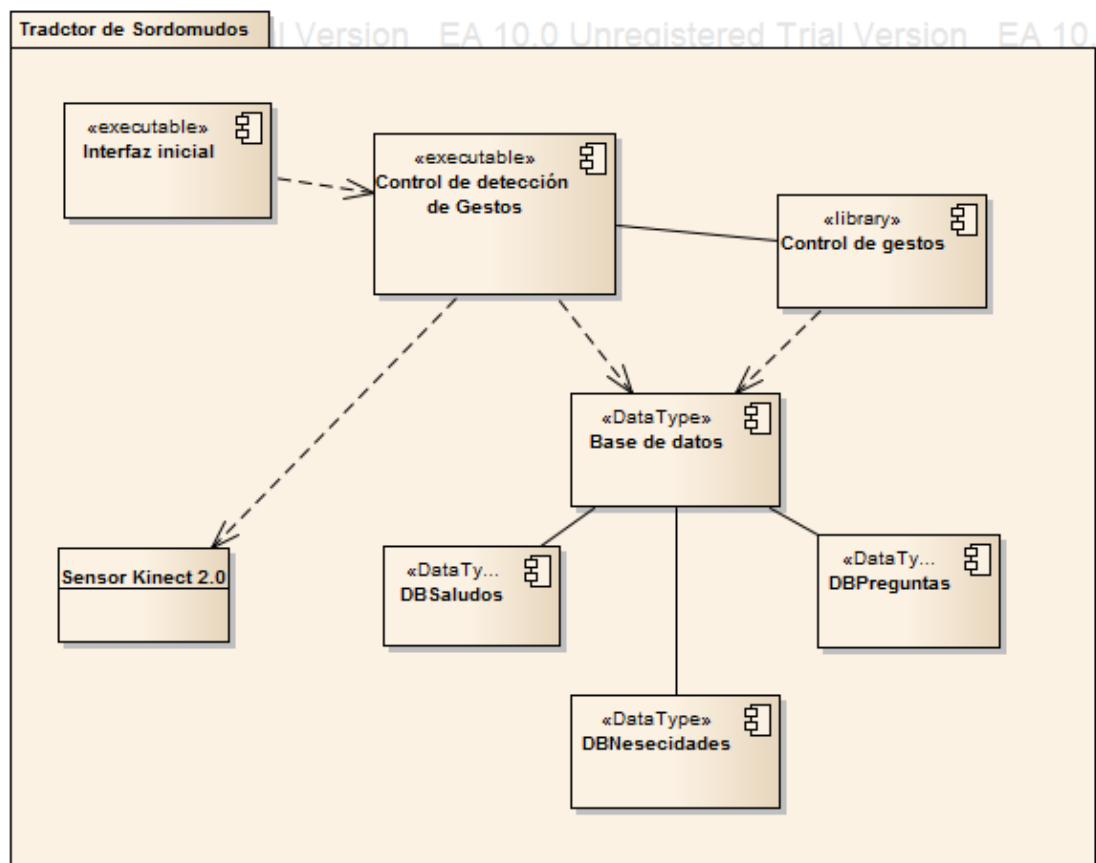


Figura 16. Diagrama de Componentes.⁶⁸

⁶⁸ Figura tomada de: creación propia.

5.2.11. Diagrama de clases

El siguiente diagrama de clases describe la estructura del sistema, muestra sus clases junto con las relaciones existentes entre ellas. Contiene información detallada de cada clase, su nombre, atributos, métodos y funciones. Para apreciar estas clases gráficamente se tiene la siguiente figura.

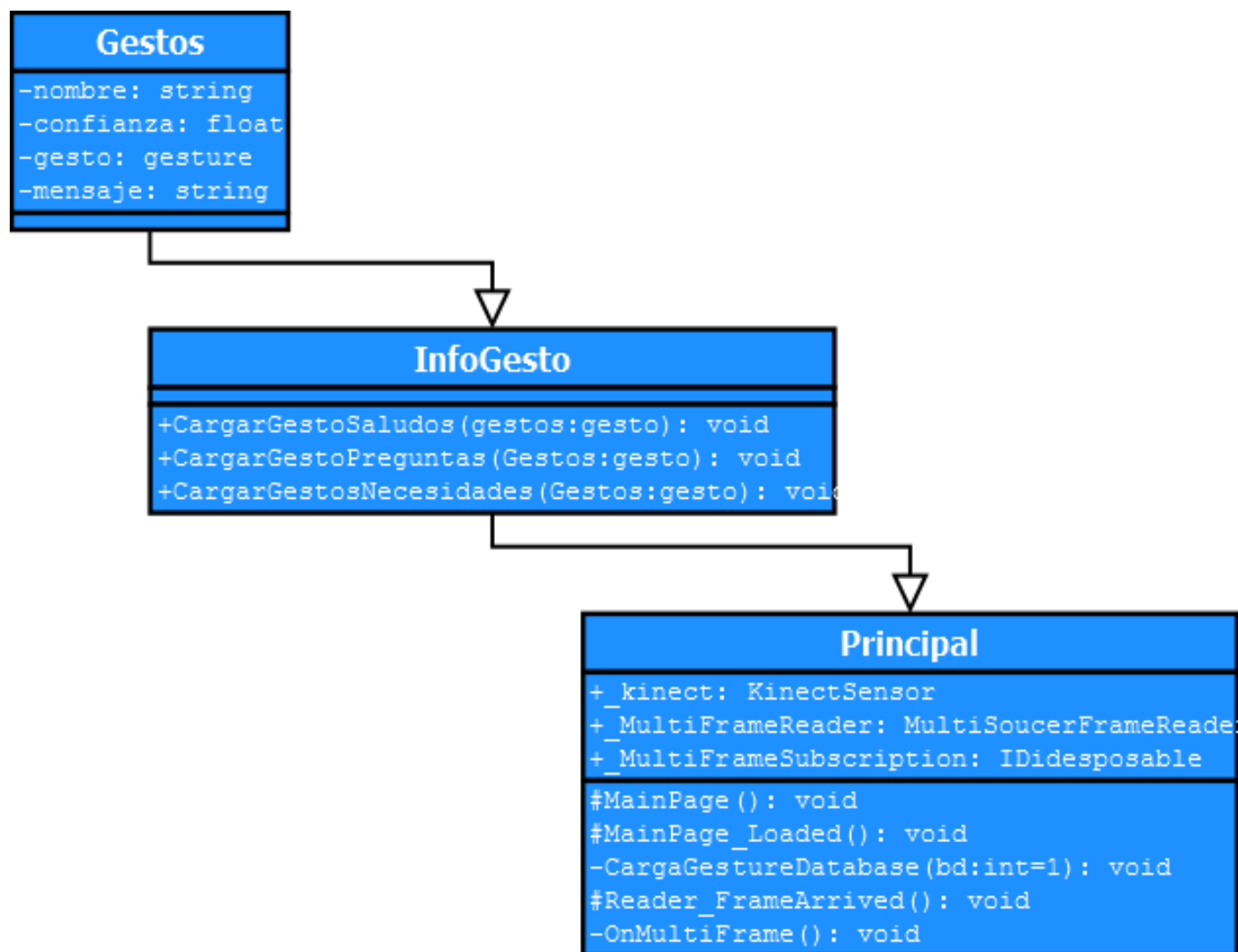


Figura 17. Diagrama de clases.⁶⁹

⁶⁹ Figura tomada de: creación propia.

5.2.12. Pruebas y análisis

Luego de haber terminado la elaboración del sistema como tal, se procede a realizar la fase de pruebas para corroborar que todo funcione como estaba previamente definido (Ver diagrama de secuencia) y de igual forma darnos cuenta si el sistema si desarrolla los procesos como se había definido durante el diseño.

En esta fase también nos podemos dar cuenta si el sistema presenta fallas en su funcionamiento o si está funcionando como se planteó en un principio, también podemos confirmar si las respuestas que el sistema genera tras recibir los diferentes estímulos proporcionados por los usuarios responden o no, de la forma correcta, determinando de esta manera en que procesos se presentan fallas, para posteriormente realizar las correcciones necesarias.

Para la primera prueba se verifican, el estado de las cámara y cada uno de los sensores del Kinect, que el terminal con Windows tenga todas las configuraciones necesarias para poder comunicarse con el sensor, y funcionalmente también la detección del cuerpo del usuario, la realización del gesto y la traducción del gesto, en donde el usuario sigue la secuencia normal para usar el sistema.

Tras realizar la primera prueba funcional del sistema me di cuenta que el reconocimiento de los patrones de gestos no era el esperado, ya que en un principio la versión inicial del sistema estaba conformada únicamente por gestos de tipo discreto, los cuales responden al estímulo al determinar el comienzo y el final de un gesto mas no interpretaba el evento del gesto como tal, es decir solo interpretaba el punto en donde comenzaba el gesto y el punto donde terminaba, lo cual generaba que algunos gestos se confundieran entre sí, puesto que habían casos en los que algún gesto “X” en el momento de estar siendo ejecutado pasaba necesariamente ya fuera por el punto de inicio o final de otro gesto “Y”. Como se muestra en las figuras 16 y 17 a continuación:



Figura 18. Realización del gesto "Buenos días"⁷⁰.



Figura 19. Realización del gesto "Buenos días", mal interpretado⁷¹.

⁷⁰ Figura tomada de: creación propia.

⁷¹ Figura tomada de: creación propia.

Lo cual postulo un problema preocupante, motivo por el cual fue necesario asociar por cada gesto discreto un gesto de tipo continuo que me permitiera controlar los datos que se generaban durante secuencia del gesto. Esta nueva estructura de control de los gestos me permitió administrar a un nivel mucho más completo la secuencia entera de los gestos, dándome la libertad de manipular las condiciones a mi voluntad, es decir, basándonos en el ejemplo anterior donde el gesto “X” y el gesto “Y”, coincidían en alguno de sus puntos iniciales o finales ya yo los iba a poder diferenciar debido a que cada uno seguía una secuencia diferente para su elaboración.

Corrección que mejoro exponencialmente el proceso de identificación de los gestos. Seguido se inició nuevamente el proceso de pruebas con la nueva configuración del sistema de la siguiente forma:

En primera medida se inicia la aplicación, luego el usuario se ubica frente al sensor Kinect a una distancia contemplada entre 0.5 y 4.5 metros, seguido a esto el usuario procede a realizar un gesto válido que se encuentre dentro de la base de datos e inmediatamente el sistema traduce el gesto realizado a un texto totalmente entendible, Pero cuando se realizó esta primera prueba la base de datos solo contaba con tres gestos únicamente y el comportamiento del sistema con una base de datos tan pequeña fue más que positivo. Como se muestra en la imagen siguiente.

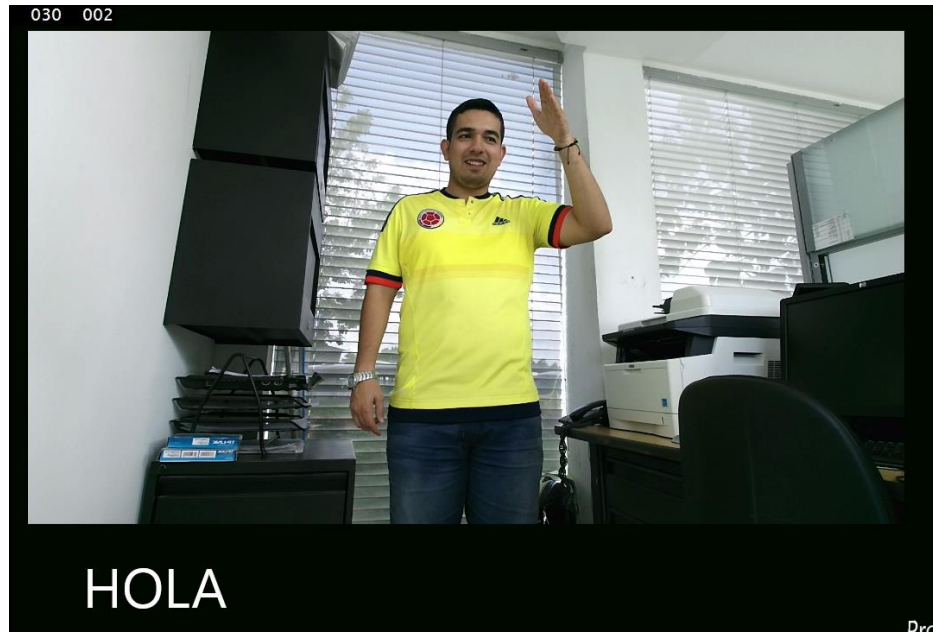


Figura 20. Realización del gesto "Hola" por otro usuario.

Durante la misma prueba el usuario también procedió a realizar gestos que no estaban habilitados en la base de datos y el resultado también fue muy positivo, porque a simple vista no ocurría nada al realizar un gesto no valido, pero en realidad el sistema al no reconocer el gesto saltaba nuevamente al estado de espera de detección de gestos, generando una interfaz totalmente amable y óptima para el usuario; A parte de esto, luego de recibir varios gestos inválidos y aun es espera de gestos después de eso, el usuario realizó nuevamente un gesto valido y de inmediato el sistema genero la traducción textual de dicho gesto.



Figura 21. Realizando gestos que no existe en las bases de datos.⁷²

No obstante luego de que esta prueba preliminar resultó exitosa se realizó una alimentación a la base de datos con la gama de gestos completa que incluía el vocabulario básico del lenguaje de señas colombiano.

Posteriormente a esta ampliación de la base de datos se inició la segunda prueba, pero se comenzaron a detectar problemas a la hora de que el usuario realizara un gesto que si estaba dentro de la base de datos, ya que al momento de que la base de datos se volvió más robusta la diferenciación entre los diferentes gestos se volvió más difícil ya que al usuario realizar un gesto había una posibilidad de que el gesto se confundiera con otro que poseía algunos atributos similares.

Por ello se decidió reestructurar la base de datos y se subdividió el vocabulario básico del lenguaje de señas colombiano en tres secciones asociadas cada una a una base de datos independiente, las cuales fueron la sección de Saludos, Necesidades y por ultimo

⁷² Figura tomada de: creación propia.

preguntas, distribuyendo los diferentes tipos de gestos entre estas tres bases de datos respectivamente, a partir de aquí se alimentaron las bases de datos, cada una con su respectivo tipo de gestos. A partir de este punto la interfaz de usuario cambio, evidenciando las tres secciones en las cuales quedo dividido el sistema así:



Figura 22. Vista preliminar de la nueva interfaz del sistema⁷³.

Finalizada esta corrección se procedió a realizar una tercera prueba en la cual también se habían habilitado gestos específicos para que el usuario pudiera navegar con total fluidez entre las tres diferentes secciones que ahora existían en el sistema. Seguido a esto el usuario procedió a realizar la secuencia normal que se debe seguir para poder traducir los gestos, se inició la aplicación, el usuario se ubicó en frente al sensor, su cuerpo fue calibrado y luego este procedió a realizar un gesto existente dentro de la base de datos saludos que es la base de datos que carga al inicio de la ejecución del sistema por defecto, obteniendo resultados muy prometedores en esa base de datos específica; Al terminar de probar la fluidez de los gestos existentes en esta base de datos el usuario realizo el gesto

⁷³ Figura tomada de: creación propia.

establecido para cambiar a la siguiente base de datos y el resultado de esta acción fue totalmente satisfactorio, el sistema cargo los datos de la nueva base de datos en tiempo de ejecución permitiéndole al usuario probar los gestos de esta base de datos, obteniendo también muy buenos resultados de los gestos existentes en esta y así posteriormente el usuario realizó el gesto determinado para pasar a la siguiente base de datos lo cual también resultó un éxito y ya dentro de la última base de datos, inicio nuevamente la inspección de los gestos existentes en la nueva base de datos, observando que todos los gestos existentes dentro de la misma respondían correctamente a la realización de los gestos realizados por él.



Figura 23. Realización del gesto para pasar de una sección a otra⁷⁴.

⁷⁴ Figura tomada de: creación propia.



Figura 24. Gesto para navegar entre las secciones hecho por otro usuario⁷⁵.

Todas las pruebas realizadas anteriormente sirvieron como fundamento para responder a una incógnita muy importante, ¿Funcionará igual con una persona de una contextura diferente?

Por lo que se realizaron con usuarios que contaban con una estatura entre 1.70 y 1.80 metros y resultaron satisfactorias, ayudando a resolver otro carácter importante ya que el volumen corporal de los dos usuarios era relativamente diferente, lo que sirvió para verificar que el dispositivo reacciona correctamente ante diferentes tipos de usuarios.

A parte de las anteriores se realizó una quinta y última prueba para verificar si existiendo dos usuarios dentro del rango de detección del sensor este se seguía comportando igual, y los resultados arrojaron que si es posible reconocer los gestos realizados por cada uno de los cuerpos independientemente siempre y cuando los cuerpos no estuvieran súper puestos, es decir, que un cuerpo no obstaculice al otro (ej.: un cuerpo frente al otro, uno de los brazos de un cuerpo frente al otro cuerpo y casos de ese tipo). También se hizo

⁷⁵ Figura tomada de: creación propia.

evidente que la traducción de los gestos era un poco más retardada en comparación a cuando se hizo la misma prueba con un solo cuerpo, es decir, los gestos no se traducían con la fluidez normal, pero era un retardo muy pequeño el cual se consideró dentro de los rangos de tiempo aceptables desde la realización de un gesto hasta su traducción. Como se evidencia en las figuras siguientes:

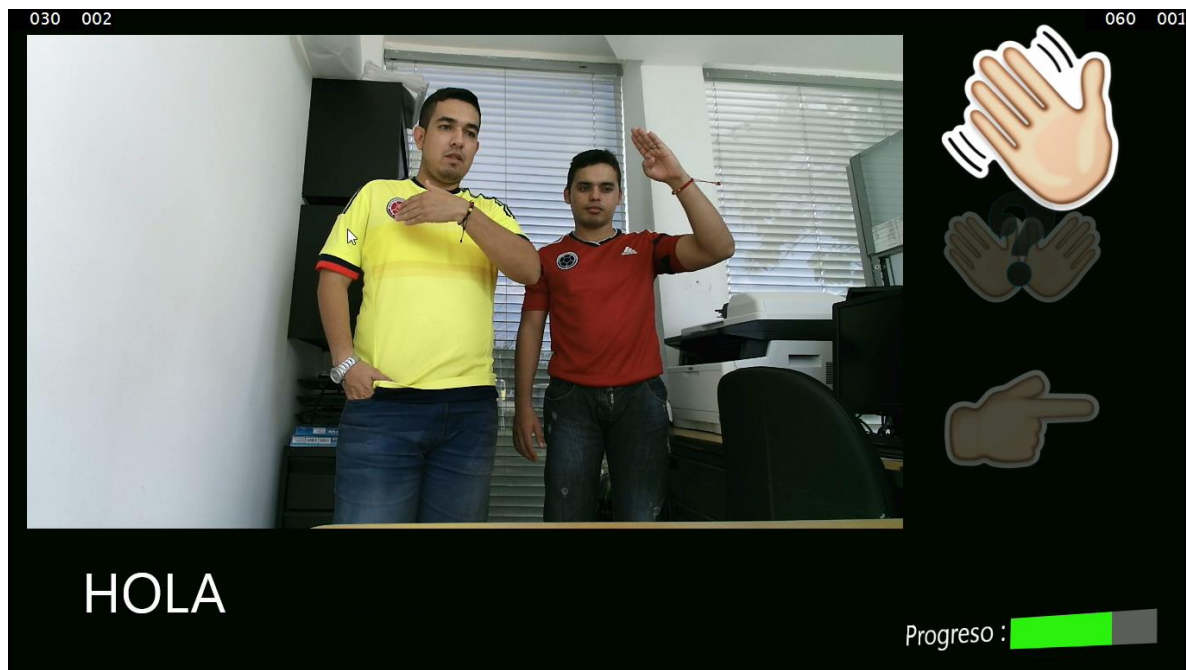


Figura 25. Realización del gesto "Hola" con 2 usuarios frente al sensor⁷⁶.

En la prueba anterior se evidencia que el usuario de rojo realizó al gesto “hola”, y que a pesar de que el otro usuario también se estaba moviendo, el sistema fue capaz de detectar el gesto.

⁷⁶ Figura tomada de: Creación propia.



Figura 26. Gesto para navegar entre secciones con dos usuarios frente al sensor⁷⁷.

Durante el desarrollo de las pruebas el sistema realizó correctamente todos los procesos especificados durante el desarrollo del proyecto, el sistema en cuestión de instantes fue capaz de reconocer y calibrar el cuerpo de cualquier usuario de una forma prácticamente imperceptible para el mismo, también le permitió a los usuarios que probaron el sistema realizar su gesto en el mismo instante en el cual se ubicaban frente al sensor, lo cual habla muy bien del sistema de reconocimiento del cuerpo de un nuevo usuario.

A la hora de la traducción de los gestos realizados por el usuario, el sistema validó de forma inmediata si el gesto existía o no dentro de las variables de programación asociadas a la base de datos que posteriormente serían empleadas como llaves para identificar cada uno de los gestos ingresando a la base de datos y cargando los datos relacionados al gesto anteriormente identificado, capturando una variable de texto existente en el algoritmo de traducción de gestos y mostrándola en la interfaz visual de la aplicación, eso en el caso de que el gesto existiera dentro de la base de datos, en los

⁷⁷ Figura tomada de: creación propia.

casos en los que el gesto no era válido o simplemente no existía dentro del contexto de la base de datos, el sistema lo validaba y como no existía en programación una variable asociada para su posterior identificación, simplemente lo descartaba y automáticamente quedaba en espera del reconocimiento del nuevo gesto, pero todos estos procesos descritos anteriormente se realizaban a una velocidad tal, que ni si quiera en la interfaz visual de usuario se alcanzaban a percibir.

La aplicación cumplió plenamente con los objetivos propuestos en el principio del desarrollo del proyecto, identificando de la manera más natural posible los gestos realizados por cualquier usuario que se ubique dentro el rango de identificación del sensor.

6. CONCLUSIONES

En muchos países alrededor del mundo se han venido instaurando acciones que permitan de una forma u otra aumentar la inclusión social de las personas con discapacidad sordo muda en todos los campos habituales de la sociedad, como trabajar, estudiar, ser atendidos en el sector salud, transportarse, entre muchas otras que para las personas del común y corriente son de lo más cotidianas, estas acciones se apoyan en las leyes y la normatividad asociada a esta problemática.

Durante el desarrollo de esta investigación se pudo evidenciar que esta problemática está siendo abordada tanto por el sector social como el sector educativo, siendo este último en el cual se han propuesto más alternativas para contribuir de manera significativa a la solución de esta problemática.

En Colombia se implementó el Plan Nacional de Desarrollo del gobierno denominado “Prosperidad para Todos” 2010 – 2014 (Ministerio de salud y protección social, 2013), el cual plantea una serie de condiciones y procesos pertinentes para garantizar la inclusión social de las Personas con Discapacidad y sus familiares. Con el diseño e implementación se busca contribuir lo más posible en mejorar la calidad de vida de estas personas.

Los dispositivos existentes se diferencian de este Traductor de sordomudos, principalmente por la fidelidad de los datos y por la capacidad que este sistema posee para traducir frases completas del vocabulario de señas colombiano, en donde la mayoría de los dispositivos desarrollados con este mismo propósito interpretan una a una las letras abecedario.

Los resultados obtenidos a la hora de la culminación del proyecto cumplieron con todos los lineamientos y requerimientos establecidos en el diseño del mismo como se ve en el apartado de pruebas.

7. REFERENCIAS BIBLIOGRAFICAS

- Bermúdez, C., Garrido, É., Canella, M., Juárez, N. L., & Argentina, U. d. (2013). Traducir en pantalla el lenguaje de señas Argentino.
- Chapala, Instituto Tecnológico Superior (ITS) de. (2010). Interprete de señas.
- DANE . (2005). *Censo General del DANE*.
- DANE. (2005). *Instituto nacional para sordos (INSOR)*.
- Daniel Betancur Betancur, Mateo Vélez Gómez & Alejandro Peña Palacio. Escuela de ingenierías de Antioquia. (2013). Desarrollo de un sistema para el reconocimiento automático del lenguaje dactilológico de señas. *Revista Ingeniería Biomédica ISSN 1909-9762 / Volumen 7 / Número 13* .
- Du, H., & To, T. (2012). Hand gesture recognition using Kinect, Boston University.
- esmsdn. (2011). Kigo el robot clasificador de residuos.
<http://www.isedkigo.blogspot.com/2012/05/integracion-de-kinect-parte-i.html>.
- FENASCOL. (2014). *Las personas sordas*. Obtenido de
http://www.fenascol.org.co/index.php?option=con_content&view=article&id=13&showall=1
- Galindo, D. (2014). *Software que permite a los sordos comunicarse sin interprete*. Obtenido de <http://elcomercio.pe/tecnologia/inventos/software-permite-sordos-comunicarse-sin-interprete-noticia>
- Incertis, I. (2006). Hand gesture recognition for deaf people interfacing in the 18th international conference on pattern recognition. *(ICPR'06) IEEE*.
- Isamar Cartagena & Katherine Fernández. (2013). *Parque explora Medellín, Vibrasor, proyecto para sordos*. Obtenido de <http://www.parqueexplora.org/educacion>
- L., R. (2009). Sistema para el reconocimiento del alfabeto de dactilológicos (tesis).
- Leal, D. (2012). *Semana. Colombiano crea traductor de lenguaje de señas para sordos*. Obtenido de <http://www.semana.com/vida-moderna/articulo/colombiano-crea-traductor-lenguaje-senas-para-sordos/353935-3>.
- Li, Y. (2012). Hand gesture recognition using kinect. *Science IEEE*.
- Microsoft. (2012). *INFORMADOR.MX*. Obtenido de Le dan voz al lenguaje de señas con tecnología de Microsoft:
<http://www.informador.com.mx/tecnologia/2012/366529/6/le-dan-voz-al-lenguaje-de-senas-con-tecnologia-de-microsoft.htm>
- Ministerio de salud y protección social. (2013). *Prosperidad para todos*.

- Organización Mundial de la Salud. (2011). *Informe mundial de discapacidad*.
- Organizacion Mundial de la Salud. (2011). *Informe mundial sobre la discapacidad*.
Obtenido de www.who.int/disabilities/world_report/2011/summary_es.pdf
- Organizacion Mundial de la Salud. (2015). Obtenido de
<http://www.who.int/mediacentre/factsheets/fs300/es/>
- Rama, A., & Tarrés, & F. (2010). método para la detección de caras basado en Integrales Difusas (Universitat Politècnica de Catalunya, Barcelona, Spain).
- Serradilla, Rodríguez, J. M., & Francisco, y. (2003). Reconocimiento automatico del lenguaje de signos a partir de datos tridimensionales correspondientes a posición y orientación de la mano capturados mediante el guante de datos P5.
- Universidad Católica "Nuestra Señora de la Asunción". (s.f.). TECNOLOGIA PARA DISCAPACITADOS, Teoría y Aplicación a la Informática.
- Viola, P., & Jones, & M. (2001). Robust Real-Time Object Detection.

ANEXO

MANUAL DE USUARIO

Sistema para la interpretación y traducción de gestos asociados a saludos, preguntas y necesidades básicas del lenguaje de señas colombiano.

A continuación se describe la secuencia de acciones o pasos que el usuario debe realizar para hacer uso del sistema Traductor de sordomudos.

- **Reconocimiento del cuerpo del usuario**

1. Una vez iniciado el sistema el usuario debe proceder a ubicarse frente al Kinect a una distancia entre 0.5 y 4.5 metros para que este se encargue de reconocerlo como un nuevo usuario. Como se ve en la siguiente figura:



Figura 27. Manual de usuario, reconocer el cuerpo.

Fuente: creación propia.

Luego de ubicarse a una distancia correcta frente al sensor, el sistema calibra las dimensiones de su cuerpo.

- **Realizar un gesto**

A continuación se mostraran los posibles gestos que pueden ser realizados por el usuario en cada una de las diferentes secciones del sistema.

Inicialmente se mostraran los posibles gestos a realizar en la sección de saludos ya que es la interfaz que carga por defecto al inicio del sistema.

- **Gestos de la sección “Saludos”**

1. El usuario puede decir hola en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:

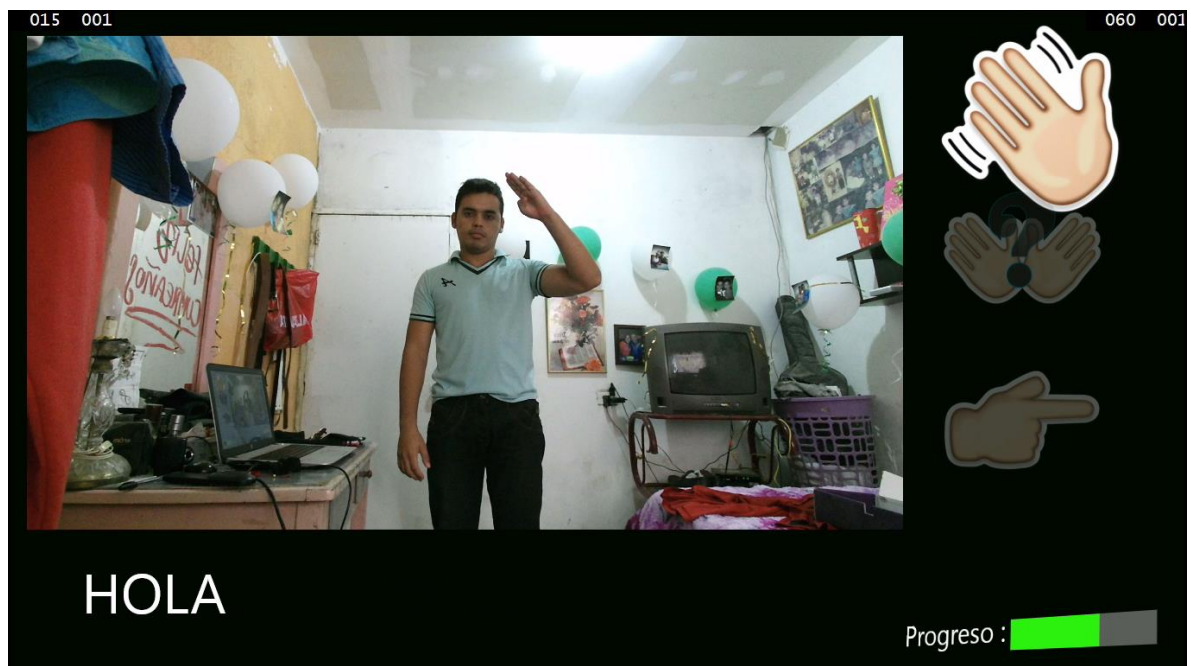


Figura 28. Manual de usuario Gesto "Hola".

Fuente: creación propia.

2. El usuario puede decir adiós en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:

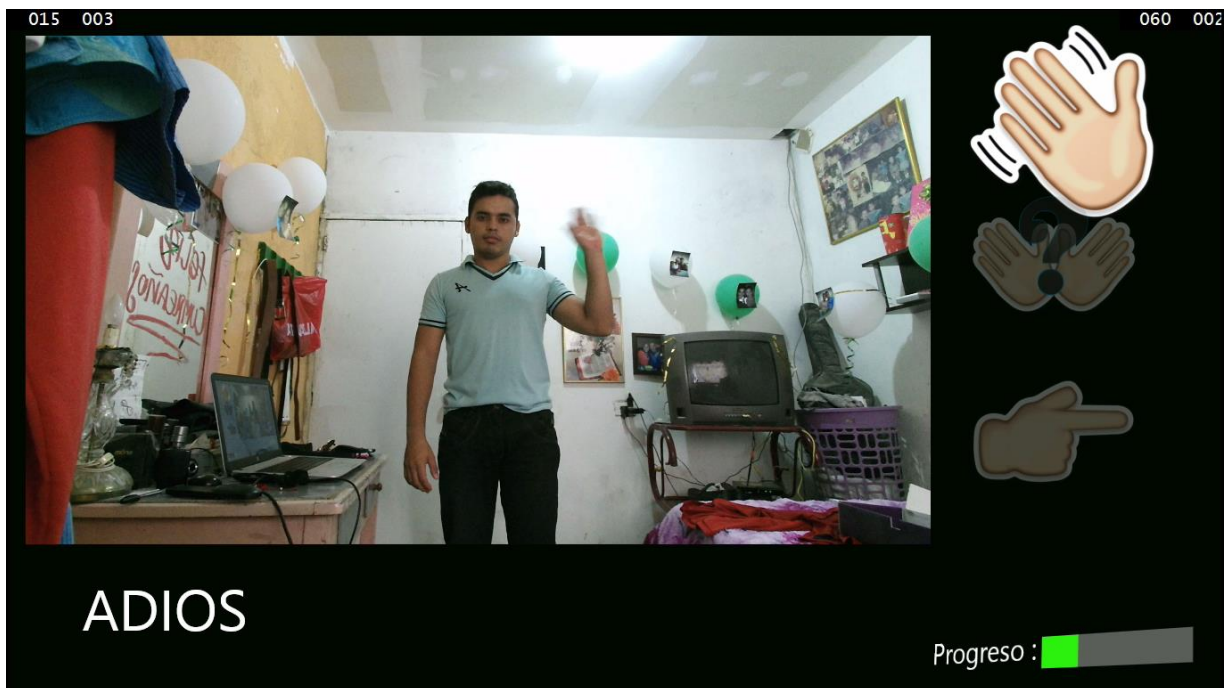


Figura 29. Manual de usuario Gesto "Adiós".

Fuente: creación propia.

3. El usuario puede decir buenos días en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:



Figura 30. Manual de usuario Gesto "Buenos días".

Fuente: creación propia.

4. El usuario puede decir buenas tardes en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:



Figura 31. Manual de usuario Gesto "Buenas tardes".

Fuente: creación propia.

5. El usuario puede decir buenas noches en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:



Figura 32. Manual de usuario Gesto "Buenas noches".

Fuente: creación propia.

Dentro de la aplicación se evidencian 3 escenarios a través de los cuales el usuario puede navegar dependiendo del tipo de gestos que desee realizar, para realizar este procesos existen dos gestos asociados, que son Navegar hacia el siguiente escenario o navegar hacia el anterior.

6. El usuario puede navegar hacia el escenario siguiente dentro de la aplicación deslizando su mano derecha dese la posición derecha hasta la posición izquierda como se muestra en las siguientes imágenes:



Figura 33. Manual de usuario Gesto "Siguiente parte 1".

Fuente: creación propia.



Figura 34. Manual de usuario Gesto "Siguiendo parte 2"

Fuente: creación propia.

7. El usuario puede navegar hacia el escenario anterior dentro de la aplicación deslizando su mano izquierda dese la posición derecha hasta la posición izquierda como se muestra en las siguientes imágenes:

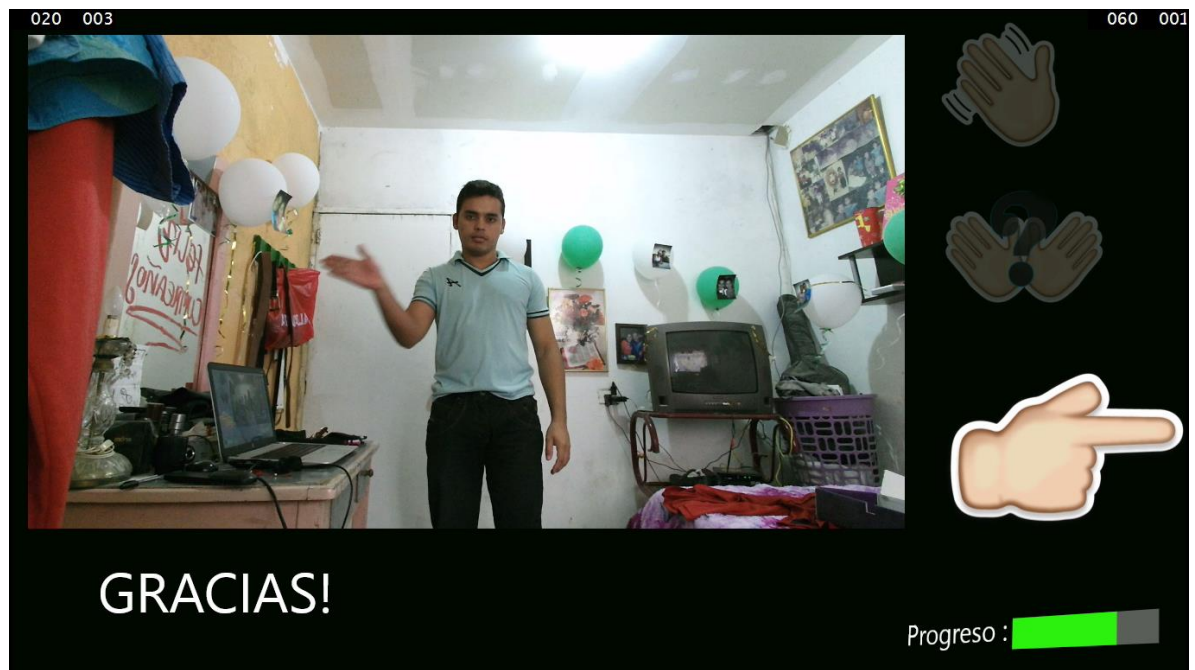


Figura 35. Manual de usuario Gesto "Anterior parte 1".

Fuente: creación propia.



Figura 36. Manual de usuario Gesto "Anterior parte 2".

Fuente: creación propia.

- **Gestos de la sección “Preguntas”**

1. El usuario puede preguntar cómo estas en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:



Figura 37. Manual de usuario Gesto "como estas".

Fuente: creación propia.

2. El usuario puede preguntar dónde está en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:



Figura 38. Manual de usuario Gesto "donde esta".

Fuente: creación propia

3. El usuario puede preguntar cuál es tu nombre en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:



Figura 39. Manual de usuario Gesto "cuál es tu nombre".

Fuente: creación propia.

4. El usuario puede preguntar por qué en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:

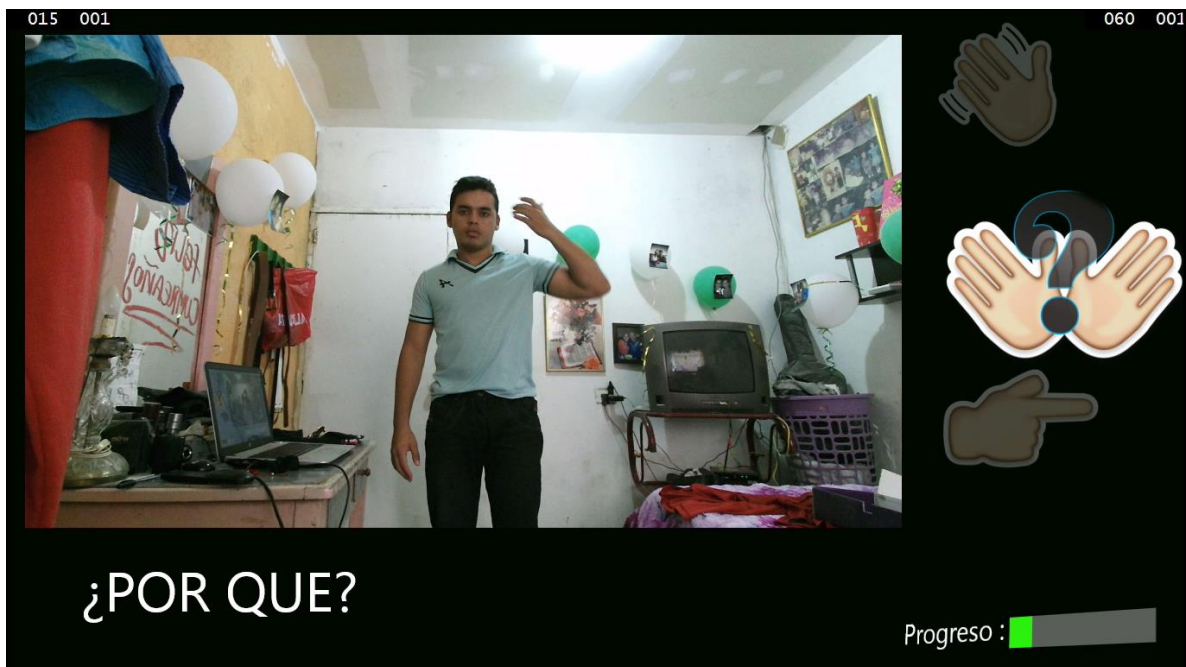


Figura 40. Manual de usuario Gesto "Por qué".

Fuente: creación propia.

5. El usuario puede preguntar que estás haciendo en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:



Figura 41. Manual de usuario Gesto "Que estás haciendo".

Fuente: creación propia.

- **Gestos de la sección “Necesidades”**

1. El usuario puede decir por favor en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:

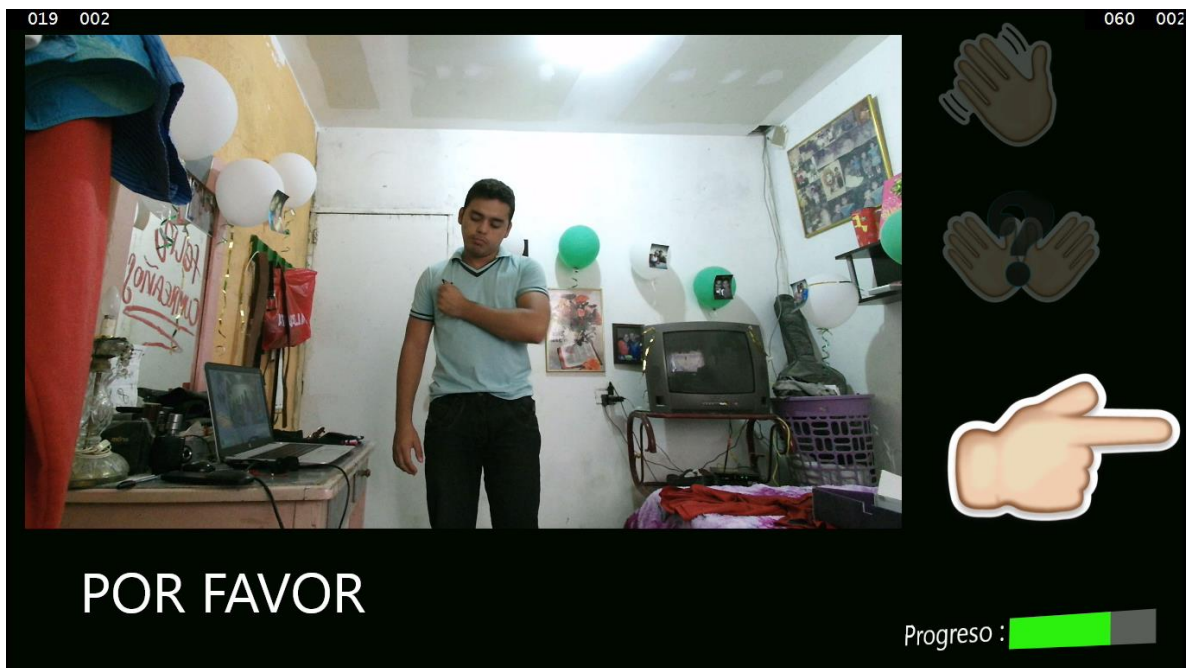


Figura 42. Manual de usuario Gesto "Por favor".

Fuente: creación propia.

2. El usuario puede decir tengo hambre en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:



Figura 43. Manual de usuario Gesto "Tengo hambre".

Fuente: creación propia

3. El usuario puede decir tengo sed en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:

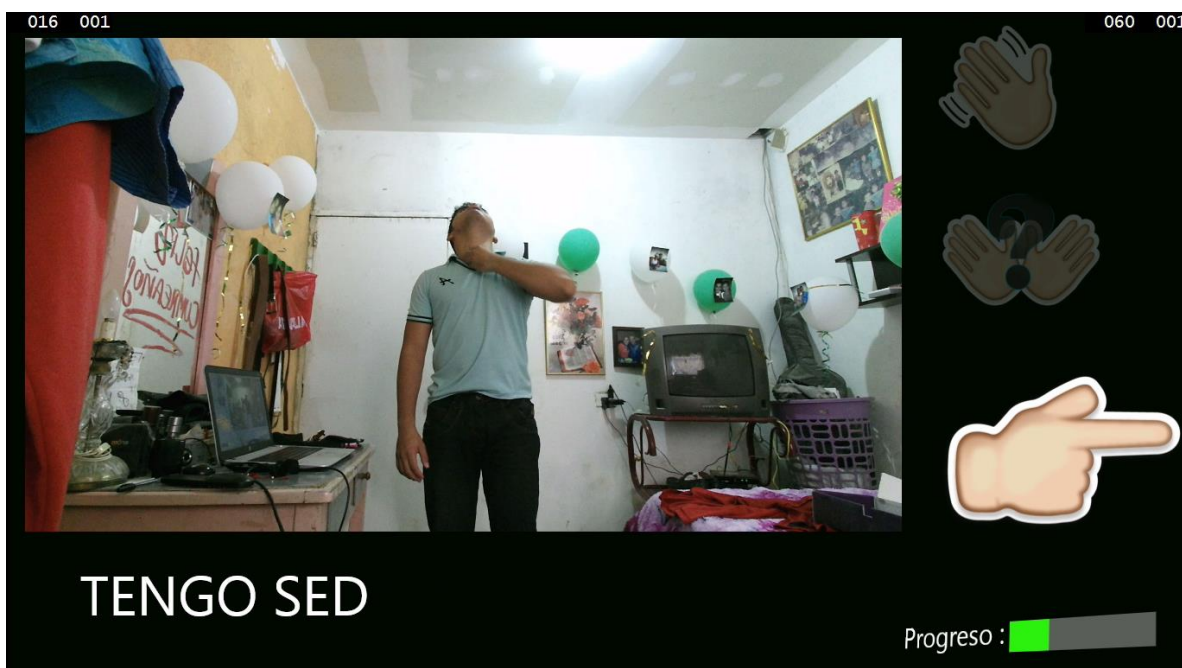


Figura 44. Manual de usuario Gesto "Tengo sed".

Fuente: creación propia

4. El usuario puede decir tengo sueño en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:

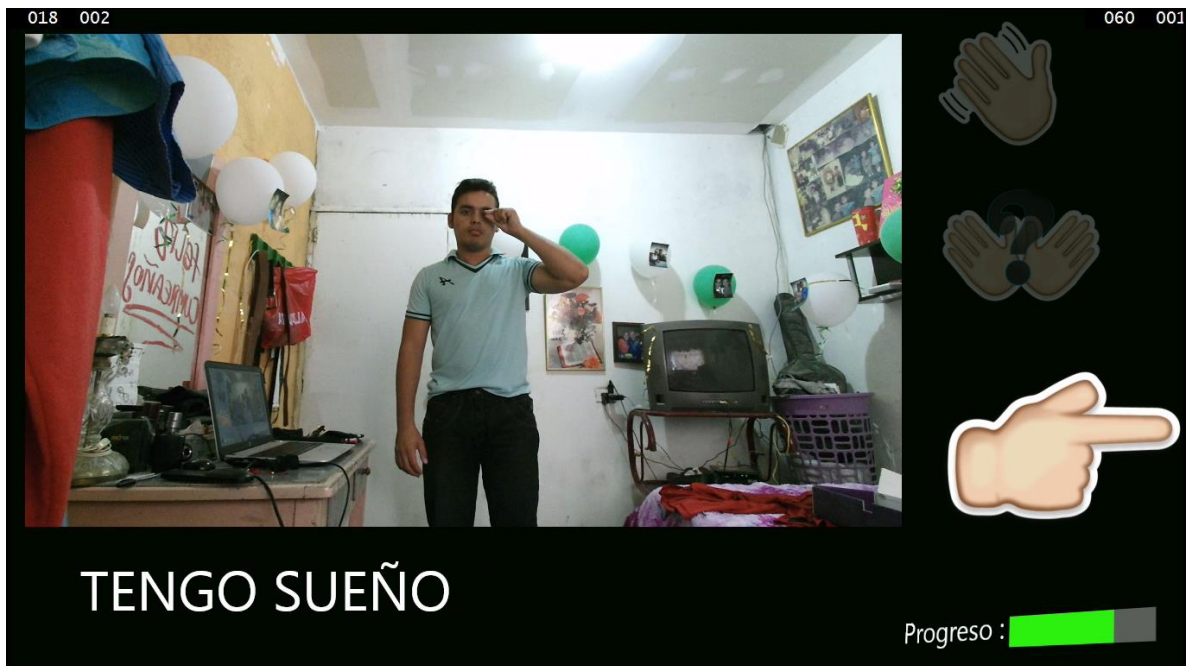


Figura 45. Manual de usuario Gesto "Tengo sueño".

Fuente: creación propia

5. El usuario puede decir gracias en su lengua de señas y el sistema reconocerá los patrones de dicho gesto y realizara la traducción. Como se ve en la siguiente figura:



Figura 46. Manual de usuario Gesto "Gracias".

Fuente: creación propia.